Polling systems

Borst, Simon Catharina

Publication date:
1994

Document Version
Publisher's PDF, also known as Version of record

Polling Systems

S.C. Borst

Polling Systems

Polling Systems

Proefschrift

ter verkrijging van de graad van doctor
aan de Katholieke Universiteit Brabant,
op gezag van de rector magnificus, prof.dr. L.F.W. de Klerk,
in het openbaar te verdedigen ten overstaan van
een door het college van dekanen aangewezen commissie
in de aula van de Universiteit
op vrijdag 4 november 1994 te 16.15 uur

door

Simon Catharina Borst

geboren te Gouda

Promotor: Prof.dr.ir. O.J. Boxma

Stellingen

behorende bij het proefschrift van

Simon Catharina Borst

Polling Systems

I

Beschouw het wachtrijmodel beschreven in sectie 2.4 van het proefschrift. Zij $\mathbf{W}$ en $\mathbf{R}$ respectievelijk de *wacht*tijd en de *verblijf*tijd van een willekeurige superklant. Laat $w(\omega) := \mathrm{E}(\mathrm{e}^{-\omega \mathbf{W}})$ en $r(\omega) := \mathrm{E}(\mathrm{e}^{-\omega \mathbf{R}})$ voor $\mathrm{Re}\,\omega \geq 0$. Dan geldt [1]

$$w(\omega) = \frac{\gamma r(\omega) - \omega r(\gamma)}{\gamma - \omega},$$

waar $r(\cdot)$ expliciet wordt gegeven door formule (2.36) van het proefschrift.

Zij $\mathbf{W}_I$ de wachttijd van een willekeurige klant in de overeenkomstige $M/G/1$ wachtrij *zonder* afhankelijkheid tussen bedieningstijden en tussenaankomsttijden, d.w.z., een gewone $M/G/1$ wachtrij met aankomstintensiteit γ en bedieningsduurverdeling met Laplace-Stieltjes getransformeerde $\gamma/(\gamma + \lambda(1 - \beta(\omega)))$. Dan geldt [1]

$$\mathrm{E}\mathbf{W} - \mathrm{E}\mathbf{W}_I = \frac{-\lambda\beta}{1 - \lambda\beta}\left[r'(\gamma) + \frac{\lambda\beta}{\gamma}\right] \leq 0.$$

De betreffende vorm van afhankelijkheid tussen bedieningstijden en tussenaankomsttijden verkleint dus de gemiddelde wachttijd, ongeacht de specifieke parameterwaarden.

[1] Borst, S.C., Boxma, O.J., Combé, M.B. (1993). An $M/G/1$ queue with customer collection. *Commun. Stat. - Stochastic Models* **9**, 341-371.

II

Langaris [2] wekt ten onrechte de suggestie als zouden zijn resultaten gelden voor een willekeurige bezetperiode. Zijn resultaten gelden feitelijk slechts voor een initiële bezetperiode [3]. Langaris geeft zich geen rekenschap van het feit dat wegens de afhankelijkheid tussen bedieningstijden en voorafgaande tussenaankomsttijden de eerste bedieningsduur in een willekeurige bezetperiode een atypische verdeling heeft. Overigens is deze verdeling niet op eenvoudige wijze te bepalen.

[2] Langaris, C. (1987). Busy-period analysis of a correlated queue with exponential demand and service. *J. Appl. Prob.* **24**, 476-485.
[3] Borst, S.C., Combé, M.B. (1992). Busy-period analysis of a correlated queue. *J. Appl. Prob.* **29**, 482-483.

III

Beschouw het volgende model. Een bediende verwerkt klanten gegenereerd door een verzameling van n typen bronnen. Laat n_j het aantal bronnen van type j zijn, $j = 1, \ldots, n$. Een type-j bron genereert een Poisson stroom van klanten met aankomstintensiteit λ_j en bedieningsduurverdeling $B_j(\cdot)$ met gemiddelde β_j en Laplace-Stieltjes getransformeerde $\beta_j(\cdot)$, $j = 1, \ldots, n$. Zij $\mathbf{V}$ de maximale hoeveelheid werk op enig moment in een willekeurige bezetperiode. Veronderstel nu dat de bedieningsduurverdelingen een exponentiële staart hebben, d.w.z., $1 - B_j(t) \sim \mathrm{e}^{\kappa_j t}$ voor $t \to \infty$ voor zekere $\kappa_j < 0$, $j = 1, \ldots, n$, en laat $\delta < 0$ zodanig dat $\delta > \max\limits_{j=1,\ldots,n} \kappa_j$. Dan geldt [4] $\Pr\{\mathbf{V} \geq v\} < \mathrm{e}^{\delta v}$ voor $v \to \infty \iff \sum_{j=1}^{n} n_j \alpha_j(\delta) < 1$, waar $\alpha_j(\delta) = \lambda_j(1 - \beta_j(\delta))/\delta$, $j = 1, \ldots, n$. In toepassingen waar de bediende een transmissiekanaal representeert, wordt de coëfficiënt $\alpha_j(\delta)$ derhalve wel de effectieve bandbreedte van een type-j bron genoemd. Veronderstel nu echter dat de bedieningsduurverdelingen geen exponentiële maar geometrische staart hebben, d.w.z., $1 - B_j(t) \sim t^{k_j}$ voor $t \to \infty$ voor zekere $k_j < 0$, $j = 1, \ldots, n$, en laat $d < 0$ zodanig dat $d > \max\limits_{j=1,\ldots,n} k_j$. Dan geldt $\Pr\{\mathbf{V} \geq v\} < v^d$ voor $v \to \infty \iff \sum_{j=1}^{n} n_j \lambda_j \beta_j < 1$. In het geval van een geometrische staart reduceert de effectieve bandbreedte dus tot de gewone verkeersintensiteit.

[4] Cohen, J.W. (1994). On the effective bandwidth in buffer design for the multi-server channels. CWI Report BS-R9406.

IV

Beschouw het volgende model. Een Poisson stroom van klanten dient te worden toegewezen aan een groep van m parallelle niet-noodzakelijkerwijs identieke bedienden. Twee bekende mechanismen hiervoor zijn de volgende.
(i). Probabilistische toewijzing volgens een kansvector $(p_1, \ldots, p_m)$, $\sum_{i=1}^{m} p_i = 1$, $p_i \geq 0$, $i = 1, \ldots, m$, d.w.z., een arriverende klant wordt met kans p_i toegewezen aan bediende i. De lange-termijn fractie van klanten toegewezen aan bediende i is dan p_i.
(ii). Patroontoewijzing volgens een vector $(i_1, \ldots, i_l)$, $i_k \in \{1, \ldots, m\}$, $k = 1, \ldots, l$, d.w.z., de $(nl + k)$-de arriverende klant wordt toegewezen aan bediende i_k, $k = 1, \ldots, l$, $n = 0, 1, 2, \ldots$. Dan is $q_i = |\{k : i_k = i\}|/l$ de lange-termijn fractie van klanten toegewezen aan bediende i.
Zij $\mathbf{W}_i'$ en $\mathbf{W}_i''$ de wachttijd van een willekeurige klant toegewezen aan bediende i onder respectievelijk probabilistische toewijzing en patroontoewijzing. Voor iedere kansvector $(p_1, \ldots, p_m)$ en $\epsilon > 0$ bestaan er dan een $l < \infty$ en een vector $(i_1, \ldots, i_l)$ zodanig dat $|p_i - q_i| < \epsilon$ en $\Pr\{\mathbf{W}_i' < t\} \leq \Pr\{\mathbf{W}_i'' < t\}$ voor alle $t \geq 0$, $i = 1, \ldots, m$.

V

Zij $C'_{\max}$ de makespan van n taken van lengte $a_1, a_2, \ldots, a_n$ op 2 parallelle identieke machines, en zij $C''_{\max}$ de makespan van $2n$ taken van lengte $a_1, a_1, a_2, a_2, \ldots, a_n, a_n$ op 4 parallelle identieke machines. Voor alle n geldt dan $C'_{\max} \leq \frac{17}{16} C''_{\max}$. Afgezet tegen vergelijkingsresultaten voor wachtrijsystemen met meerdere bedienden, suggereert deze factor dat zogeheten *resource pooling* in een stochastische context eerder zoden aan de dijk zet dan in een deterministische setting.

VI

Anders dan de gedachte achter zogeheten *load balancing* suggereert, is het in wachtrijsystemen met meerdere bedienden in het algemeen *niet* optimaal om de werklast gelijkelijk over de bedienden te verdelen.

Zie bladzijde 153–154 van het proefschrift.

VII

Al te gewiekste klanten (zoals bijvoorbeeld op zaterdagmiddag in de supermarkt aan te treffen) kunnen behalve de meer lijdzame lotgenoten ook de wachtrij-analyticus het leven soms behoorlijk zuur maken.

VIII

Het bepleiten van collectieve maatregelen ter bevordering van gedrag dat uit maatschappelijk oogpunt wenselijk is maar op individuele basis niet lonend, en datzelfde gedrag vervolgens in afwachting van die maatregelen zelf nalaten, getuigt eerder van realiteitszin dan van hypocrisie.

IX

De slogan: "50 km: de uiterste limiet" (zoals te lezen op plakkaten langs de weg) wekt ten onrechte de suggestie als zouden er ook nog andersoortige limieten bestaan.

X

Nu autorijden nog altijd niet bij wet als milieudelict is aangemerkt, ontstaat helaas de indruk dat het aanwenden van de opbrengst van de accijnsverhogingen voor de bouw van extra cellen slechts is bedoeld om het publieke ongenoegen over een lastenverzwaring te neutraliseren door te appelleren aan een ander wijdverbreid gevoel van onbehagen.

XI

Het is vrij aangenaam leven in een land waar gemiddeld meer promovendi samen op een kamer zitten dan gedetineerden in een cel.

XII

Een goede stelling is niet noodzakelijkerwijs juist.

Dankwoord (Acknowledgements)

Dit proefschrift is het resultaat van het promotie-onderzoek dat ik gedurende de afgelopen vier jaar op het CWI in Amsterdam heb verricht. Ik ben het LNMB erkentelijk voor de oio-positie die het heeft geboden. Verder hebben diverse personen aan de totstandkoming van het proefschrift bijgedragen, van wie ik enkelen graag met name zou willen noemen.
Het is een groot voorrecht en plezier geweest door Onno Boxma te worden begeleid. Ik heb sterk profijt gehad van zijn deskundigheid en heb zijn voortdurende betrokkenheid bijzonder op prijs gesteld. De hoofdstukken 3, 6, en 7 van het proefschrift bevatten de resultaten van onze samenwerking; ook in het overige promotie-onderzoek heb ik talloze malen mogen profiteren van zijn stimulerende opmerkingen en kritische kanttekeningen.
Niet minder is het een genoegen geweest gedurende de afgelopen vier jaar met Marco Combé kamer en oio-ervaringen te mogen delen. Onze samenwerking en gesprekken, over wachttijdtheorie maar ook over talrijke alledaagse onderwerpen, hebben de tijd die ik op het CWI heb doorgebracht bijzonder veraangenaamd.
Gezamenlijk onderzoek met Hanoch Levy legde de basis voor hoofdstuk 6 van het proefschrift; hoofdstuk 10 berust op de resultaten van gezamenlijk onderzoek met Rob van der Mei. Ger Koole en Jacques Resing maakten verscheidene nuttige opmerkingen ter verduidelijking van het bewijs van Lemma 5.B. Rob van den Berg deed diverse bruikbare suggesties ter verbetering van het bewijs van Lemma 8.B.
Adri Steenbeek en Peter de Waal toonden zich keer op keer behulpzaam om onwillige computers in het gareel te brengen. Lieke van den Eersten-Schultze was zo vriendelijk om het taalgebruik in de eerste twee hoofdstukken van het proefschrift te toetsen aan het gangbare Amerikaans-Engels.
Hun allen ben ik dankbaar.

Amsterdam, augustus 1994

Sem Borst

Contents

Chapter 1

Introduction

1.1 Background and motivation

Queueing phenomena may be observed in several real-life situations when service facilities (counters, elevators, telephone lines, traffic lights) cannot immediately render the amount or the kind of service required by their users. Also, at byte level in modern data-handling technologies (communication systems, computer networks) queueing phenomena may be encountered which are typically less visible but the effects of which at user level are usually not less serious. Quite often such congestion effects may be adequately studied by mathematical methods from queueing theory. Adopting the abstract terminology from queueing theory, the main entity in a queueing model is a *queue* or *station* where *customers* arrive which require some amount of service from a *server*. Typically, queueing models are of a stochastic nature, in the sense that the duration of the interarrival and service times of the successive customers is not exactly specified but described in terms of probability distributions. The stochastic nature of queueing models reflects the fact that in most of the applications it is intrinsically random or uncertain at what time demand occurs for what amount of service.

The classical model in queueing theory consists of a *single* queue attended by a *single* server. Single-server single-queue models have been studied extensively in the literature, cf. Cohen [73] for a rigorous treatment of the main analytical results. In several situations the traditional single-server single-queue models have proven to be very successful in accurately predicting waiting times, queue lengths, and buffer overflow probabilities. However, in most of the recent applications the parallel or distributed character of the service facilities involves queueing models with multiple servers, multiple queues, or both.

This thesis is primarily devoted to queueing models with *multiple* queues attended by a single server, visiting the queues one at a time, cf. Figure 1.1. Moving from one queue to another, the server typically incurs a non-negligible switch-over time. Such single-server multiple-queue models are commonly referred to as *polling* models. The term 'polling' originates from the polling data link control scheme, in which a central computer cyclically polls the terminals on a communication link to inquire whether they have any data to transmit. When a terminal completes the transmission of data, the data link may be used for some system overhead, and then the central computer polls the next terminal. In the associated polling model the server represents the central computer, the queues correspond to the terminals, the customers represent the messages, and the switch-over time corresponds to the system overhead. In a broader perspective, polling models may arise in situations in which there are multiple customer classes sharing a common resource which is available to only one customer class at a time. In those situations, changing from one customer class to another usually involves a non-negligible overhead.

Stimulated by a wide variety of applications, polling models have been extensively studied in the literature, cf. Takagi [174], [175], [176] for a series of comprehensive surveys. In this thesis we provide a generalization and unification of the main exact distributional results available for polling models, present a detailed analysis of various extensions, and discuss several optimization issues. One of the main extensions concerns *multiple-server* polling models, which are of considerable practical relevance. So far, however, they have received remarkably little attention in the literature, perhaps because of the combined mathematical difficulties arising in multiple-queue and multiple-server models.

The remainder of the chapter is organized as follows. In Section 1.2 we describe the main applications of polling models in communication systems, computer networks and traditional fields of engineering like maintenance, manufacturing, and transportation. The wide diversity in applications is reflected in the numerous variants of polling models considered throughout the past decades, mostly focusing on the technologies emerging in the respective periods of time. It is however not in the scope of the thesis to present an encyclopedic categorization of the plethora of polling models considered in the literature. Instead, in Section 1.3 we provide rather a global classification, by identifying some fundamentally distinguishing features in the spectrum of polling models. In Section 1.4 we survey the state of the art in the *analysis* of polling systems. Rather than covering all technical details, we intend to illuminate the main concepts in the analysis of polling systems, which contribute to putting the thesis in the right perspective. In Section 1.5 we review the state of the art in the *optimization* of polling systems. Again, we seek to identify the main developments in the optimization of polling systems, rather than exhaustively address all the topics raised. In Section 1.6 we give an overview of the main results presented in the remainder of the thesis.

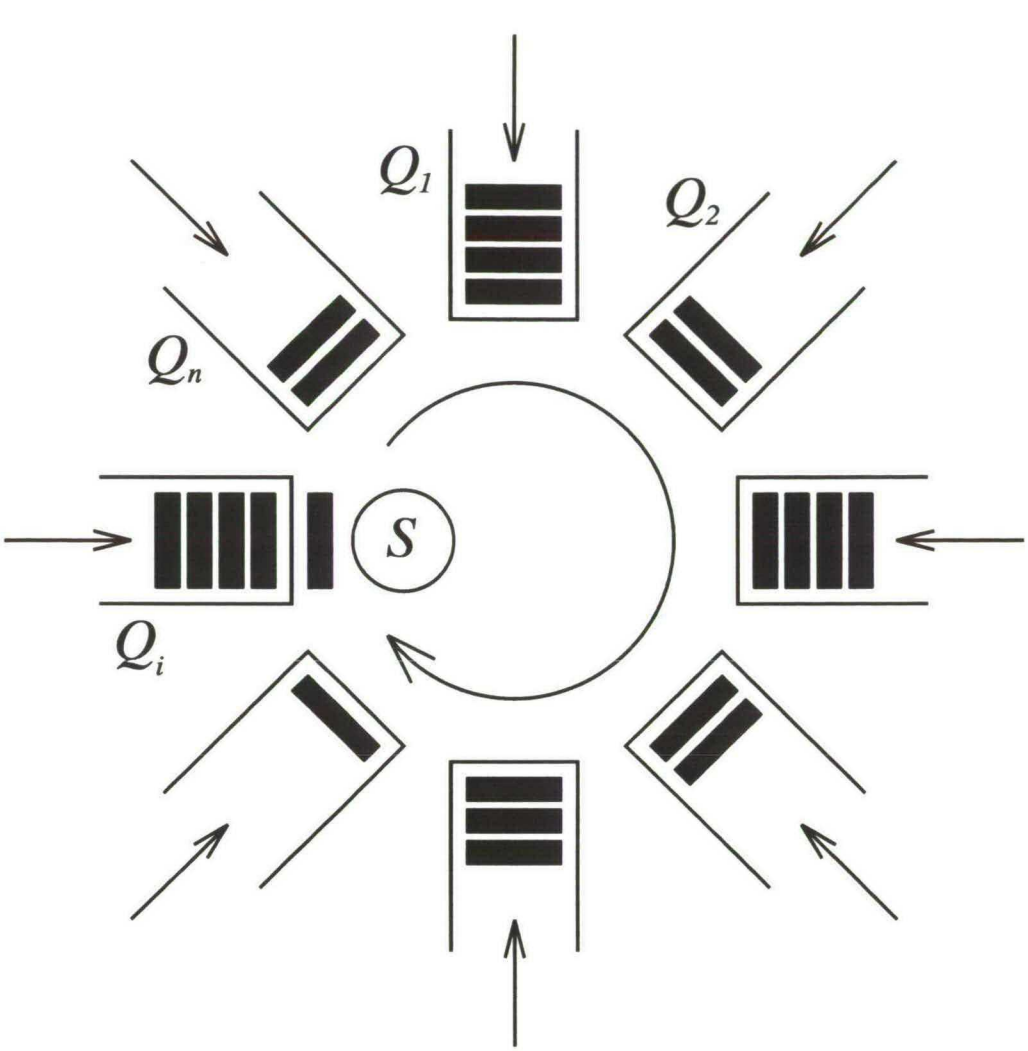

FIGURE 1.1. The basic polling system.

1.2 APPLICATIONS OF POLLING MODELS

In this section we describe the main applications of polling models in communication systems, computer networks, and traditional fields of engineering like maintenance, manufacturing, and transportation. For extensive surveys on the applications of polling models we refer to Levy & Sidi [141], Grillo [112], and Takagi [173] (the latter two surveys focusing on computer-communication systems).

Computer-communication systems
For reasons of flexibility and efficiency, modern computer systems mostly have a distributed and parallel structure. Consider e.g. a Local Area Network (LAN), consisting of a number of computers or stations interconnected by a common communication medium for exchanging packetized messages. For controlling the medium access in a LAN, one either needs a mechanism to resolve conflicts arising when more than one station starts to transmit simultaneously, or one needs a protocol to avoid such conflicts by giving only one station at a time permission to transmit data packets. The performance evaluation of the latter category, the so-called conflict-free medium access mechanisms, has greatly stimulated the research in the area of polling models. Adopting the polling terminology, the server represents the right of transmission, the queues correspond to the stations, and the customers represent the data packets. In practice, several versions of conflict-free medium access protocols are known.
One variant is the *token* ring, i.e., there is an explicit or implicit token circulating on the communication ring, representing the right of transmission. When a station receives the token, it may start transmitting packets. As soon as the station finishes transmitting, it passes the token to the next station. So holding the token corresponds to utilizing the server.
Another variant is the *slotted* ring, i.e., the communication ring is subdivided into time slots of the size of a single packet, circulating at constant speed. When a station sees an empty slot pass by, it may put a packet in it. In case of destination release the receiving station subsequently removes the packet from the slot, while in case of source release the transmitting station empties the slot again. So occupying a slot corresponds to utilizing a server.
A slotted ring may be viewed as a multiple-server polling system (unless there is only a single slot). A token ring is in fact a single-server polling system, but the stations may happen to be interconnected by multiple token rings rather than only a single token ring.

Maintenance, manufacturing, transportation
In the first polling study that appeared in the open literature, Mack, Murphy, & Webb [148] considered a situation in which a patrolling repairman cyclically inspects a number of machines, checks whether or not a failure occurred, if so repairs the machine, and then moves to the next machine. In the associated polling model the server represents the repairman, the queues correspond to the machines, and the customers represent the possible breakdowns. Königsberg & Mamer [130] studied a similar model in which an operator at a fixed position serves a number of storage locations on a rotating carrousel conveyor. Models with several independent rotating carrousels have also been considered, cf. Kim & Königsberg [127], Bunday & El-Badri [61].
There are also various applications in manufacturing environments. Consider e.g. a flexible manufacturing system, in which a machine periodically changes over from performing one type of operations to another. Here the server represents the machine and the queues correspond to the various types of operations.

A similar application is multi-product economic lot scheduling, cf. Sarkar & Zangwill [161].
Further there are applications in transportation networks. Consider e.g. a material handling system, in which a vehicle transfers loads from one machining center to another, cf. Bozer [55]. Here the server represents the vehicle, the queues correspond to the machining centers, and the customers represent the loads. Similar applications are public transport systems, mail delivery, and elevator facilities, cf. Gamse & Newell [106], [107].
A last application that is worth mentioning is the control of traffic lights. In polling terms the stream that is being given green light corresponds to the queue receiving service.

1.3 Model description

As described in the previous section, polling models find a variety of applications in communication systems, computer networks, and fields like maintenance, manufacturing, and transportation. The wide diversity in applications is reflected in the numerous variants of polling models considered throughout the past decades, mostly focusing on the technologies emerging in the respective periods of time. However, as stated before, it is not in the scope of the thesis to present an exhaustive taxonomy of the abundance of polling models considered in the literature. Instead, in this section we provide rather a global classification, by identifying some fundamentally distinguishing features in the spectrum of polling models. For a series of comprehensive surveys of the overwhelming variety of polling models considered in the literature we refer to Takagi [174], [175], [176].

The basic model
A polling model basically consists of multiple queues, $Q_1, \ldots, Q_n$, attended by a single server S. Customers arriving at Q_i are also referred to as type-i customers, $i = 1, \ldots, n$.
As usual in the recent polling literature, in the sequel the queues are always assumed to have infinite buffer capacity. In some applications (manufacturing, transportation) the inherent finiteness of the buffer capacity may play a major role in the operation of the system. However, in most of the recent applications (communication systems, computer networks) the finiteness of the buffer capacity only tends to have a minor influence on the performance of the system. In those situations the assumption of infinite buffer capacity is quite often a reasonable idealizing approximation, which facilitates the analysis considerably.

In addition to the description of the physical layout of the system, a model description essentially includes two main facets. First, the specification of the input to the system, i.e., the rules governing the duration of the interarrival,

service, and switch-over times. Second, the description of how the input is handled by the system, i.e., the rules controlling the server action.
We first consider the arrival, service, and switch-over processes. We focus on continuous-time models, i.e., the interarrival, service, and switch-over times are assumed to be continuous-valued stochastic variables. Although occasionally some subtleties may be involved, most of the results for continuous-time models carry over to discrete-time models.

The arrival process
Type-i customers arrive at generally distributed interarrival times $\mathbf{A}_i$, having distribution $A_i(\cdot)$ with Laplace-Stieltjes Transform (LST) $\alpha_i(\cdot)$, first moment α_i, and second moment $\alpha_i^{(2)}$, $i = 1, \ldots, n$. Denote by $\lambda_i := 1/\alpha_i$ the arrival rate at Q_i, $i = 1, \ldots, n$. The total arrival rate is $\lambda := \sum_{i=1}^{n} \lambda_i$.
As usual in the polling literature, in the sequel customers are always assumed to arrive according to Poisson processes, unless specified otherwise. In the absence of detailed information on the characteristics of the arrival process, the assumption of Poisson arrival processes is quite often a reasonable idealizing approximation, which facilitates the analysis considerably.
We focus here on models with single arrivals, i.e., customers are assumed to arrive one by one. Most of the results may however be generalized to models with batch arrivals.

The service process
Type-i customers require generally distributed service times $\mathbf{B}_i$, having distribution $B_i(\cdot)$ with LST $\beta_i(\cdot)$, first moment β_i, and second moment $\beta_i^{(2)}$, $i = 1, \ldots, n$. Define $\rho_i := \lambda_i \beta_i$ as the traffic intensity at Q_i, $i = 1, \ldots, n$. The total traffic intensity is $\rho := \sum_{i=1}^{n} \rho_i$.

The switch-over process
Moving from Q_i to Q_j, the server incurs a generally distributed switch-over time $\mathbf{S}_{ij}$, having distribution $S_{ij}(\cdot)$ with LST $\sigma_{ij}(\cdot)$, first moment s_{ij}, and second moment $s_{ij}^{(2)}$, $i, j = 1, \ldots, n$. As usual in the polling literature, in the sequel switch-over times are always assumed to depend only on the previous queue visited or the next queue to be visited, i.e., $\mathbf{S}_{ij} = \mathbf{S}_i$ or $\mathbf{S}_{ij} = \mathbf{S}_j$, $i, j = 1, \ldots, n$, respectively. Thus the distribution of the total switch-over time incurred during a tour along the queues has LST $\sigma(\cdot) := \prod_{i=1}^{n} \sigma_i(\cdot)$, first moment $s := \sum_{i=1}^{n} s_i$, and second moment $s^{(2)} := \sum_{i=1}^{n} \sum_{j=1}^{n} s_i s_j + \sum_{i=1}^{n} (s_i^{(2)} - s_i^2)$. To avoid ambiguity, in the sequel $\mathbf{S}_i$ always corresponds to the switch-over time incurred when swapping *out of* Q_i, unless specified otherwise.

Remark 1.3.1
The successive interarrival, service, and switch-over times are implicitly assumed to be independent. In addition, the arrival, service, and switch-over processes are assumed to be mutually independent. In some of the polling applications, however, neither of these assumptions is very realistic. Not only bursty traffic due to packetizing of messages, or due to alternating on/off phases of sources, but also collection or reservation mechanisms for transmission of messages may cause dependence in the arrival and service processes, e.g., dependence between consecutive interarrival times, between consecutive service times, or between the interarrival and service time of a customer. As Combé [76] demonstrates, quite often such dependence structures may be adequately modeled by a batch Markovian arrival process (BMAP), which is a direct generalization of the batch Poisson arrival process. In the BMAP the arrival process is governed by an underlying Markov chain, which in the case of an ordinary Poisson process has only a single state, cf. Lucantoni [147]. The $BMAP/G/1$ queue may be numerically analyzed by the matrix-geometric method, cf. Lucantoni [146], Neuts [154]. To the best of the author's knowledge, the $BMAP_i/G_i/1$ polling model has not yet been studied.
In the polling literature it is almost exclusively assumed that the arrival, service, and switch-over processes are also independent of the state of the system. As a rare exception, Boxma & Kelbert [46] consider a polling system in which customers arrive at Q_i according to a Poisson process of rate λ_{ij} when the server is at Q_j. Bozer & Srinivasan [55] analyze a model in which the switch-over time depends on the state of the *previous* queue visited; Ferguson [92], [91] studies a model in which the switch-over time may depend on the state of the *next* queue to be visited. Models with state-dependent service times are also conceivable.

□

Remark 1.3.2
With regard to the customer behavior, it is almost exclusively assumed in the polling literature that customers from some external infinite source arrive at some queue, wait for some time, receive some amount of service, and then leave the system. As a rare exception, Sidi & Levy [166] and Sidi, Levy, & Fuhrmann [167] study an *open* polling network in which customers, after receiving service at Q_i, either move to Q_j with probability r_{ij}, or leave the system with probability $1 - \sum_{j=1}^{n} r_{ij}$. In a manufacturing setting customer routing may arise when parts successively undergo service in a number of stages, e.g., drilling holes of different type, painting in different colors. In the context of communication networks customer routing may occur when a station that receives a faulty message sends a negative acknowledgement to the station that transmitted the message to indicate that the message has to be retransmitted. Altman & Yechiali [9] analyze a *closed* polling network in which customers (belonging to

a permanent population), after receiving service at Q_i, move to Q_j with probability r_{ij} (with $\sum_{j=1}^{n} r_{ij} = 1$). As a generalization of customer routing, Levy & Sidi [141] describe a model with customer branching in which departures may trigger concurrent arrivals to the system.

□

Remark 1.3.3
As usual in the polling literature, we focus here on a model with a *discrete* structure, i.e., a finite number of distinct queues. Letting the number of queues tend to infinity, with the traffic intensity ρ fixed, we obtain a *continuous* polling model. Several models have been analyzed in which the server travels around a circle, on which customers arrive according to a uniform Poisson process, cf. Bisdikian & Merakos [20], Coffman & Gilbert [69], Fuhrmann & Cooper [102], Kroese & Schmidt [132]. Recently also various models have been considered in which the server traverses a graph or a region of higher dimension, or in which customers do not necessarily arrive according to uniform Poisson processes, cf. Altman & Foss [6], Bertsimas & Ryzin [17], Coffman & Stolyar [72], Kroese & Schmidt [133].

□

Remark 1.3.4
As usual in the polling literature, for now, we focus on a model with a *single* server. In the last chapters of the thesis we consider polling models with *multiple* servers.

□

We now consider the rules controlling the server action. A scheduling strategy is a collection of decision instructions for determining the server action at any given time. Occasionally a scheduling strategy will also be referred to as a scheduling discipline, a polling strategy, or a polling policy. A scheduling strategy prescribes whether the server S should serve (which customer), switch (to which queue), or idle. Those decisions are made based on some partial knowledge of the state of the system (queue lengths, past arrival patterns) and on past decisions.
Although a scheduling strategy in principle may be arbitrarily involved, it mostly decomposes into three separate control mechanisms, viz.:
i. the routing policy: in which order should S serve the queues;
ii. the service policy: while at a queue, which number of customers should S serve;
iii. the service order: while at a queue, in which order should S serve customers.

We now successively describe the main variants of these three control mechanisms.

The routing policy
The routing policy prescribes in which order S should visit the queues. In the traditional cyclic polling model the server visits the queues in a strictly cyclic order, i.e., $Q_1, \ldots, Q_n, Q_1, \ldots, Q_n, \ldots$.
One obvious generalization of strictly cyclic polling is *periodic* polling, introduced in Kruskal [134] and revisited in Eisenberg [84], Baker & Rubin [15] and Boxma, Groenendijk, & Weststrate [45]. In periodic polling, the server visits the queues in a fixed order, listed in a *polling table* of some size m, i.e., a vector of length m with components in $\{1, \ldots, n\}$. An important special case of periodic polling is *scan* polling or *elevator* polling, in which the server visits the queues in the order $Q_1, \ldots, Q_n, Q_n, \ldots, Q_1, \ldots$. Another special case of periodic polling is *star* polling, in which the server visits the queues in the order $Q_1, Q_2, Q_1, Q_3, Q_1, \ldots, Q_{n-1}, Q_1, Q_n, \ldots$.
Another natural generalization of strictly cyclic polling is *Markovian* polling, introduced in Boxma & Weststrate [53]. In Markovian polling, the server visits the queues according to a discrete-time Markov chain with state space $\{1, \ldots, n\}$, i.e., the server is routed from Q_i to Q_j with probability p_{ij}, $i, j = 1, \ldots, n$. A special case of Markovian polling is *random* polling, i.e., $p_{ij} = p_j$, $i, j = 1, \ldots, n$, analyzed in Kleinrock & Levy [128]. Mixtures of periodic polling and Markovian polling are also conceivable.
All above policies are static in the sense that the routing decisions are made independently of the state of the system, so that the sequence of the queues visited is also independent of the input sequence to the system. In dynamic policies the routing decisions are made based on some partial knowledge of the state of the system (queue lengths, past arrival patterns) and on past decisions, e.g., the server may be instructed to serve the longest queue. Evidently, in principle the performance of the system may improve substantially by using such information in making the routing decisions. However, gathering such information and implementing a sophisticated routing policy may involve a considerable communication overhead and complicate the operation of the system significantly. Therefore in practice dynamic policies are not necessarily preferable to static policies.

The service policy
While at a queue, the service policy (or strategy, or discipline) prescribes which number of customers S should serve. There are four classical service disciplines.
I. Exhaustive service.
Under exhaustive service, the server continues to work until the queue becomes empty. Customers that arrive during the course of the visit, are served in the current visit.
II. Gated service.
Under gated service, S serves only the customers that were present at the start

of the visit. Customers that arrive during the course of the visit, are served in the next visit.

III. Limited service.

Under k-limited service, the server continues to work until either a prespecified number of k customers have been served, or the queue becomes empty, whichever occurs first. There are two versions of limited service: gated-limited or exhaustive-limited service, depending on whether or not S only serves the customers that were present at the start of the visit.

IV. Decrementing service.

Under k-decrementing service, the server continues to work until either there are a prespecified number of k customers less present than at the start of the visit, or the queue becomes empty, whichever occurs first. 1-Decrementing service is commonly referred to as semi-exhaustive service.

There are also numerous probabilistic hybrids of the four classical service disciplines. To give a systematic overview, we now define a family of service disciplines which operate as follows. If there are m_i customers present at the start of the visit to Q_i, then a (random) number $\mathbf{L}_i(m_i)$ of them qualify for service. Customers arriving during the visit to Q_i qualify for service with probability p_i. The server continues to work until either a (random) number of $\mathbf{K}_i(m_i)$ customers have been served, or there are no customers left that qualify for service, whichever occurs first.

Service disciplines with $p_i = 1$ and $p_i = 0$ are frequently referred to as exhaustive-type and gated-type policies, respectively, cf. Boxma [39], Levy & Sidi [141], Levy, Sidi, & Boxma [142]. Disciplines with $\mathbf{K}_i(m_i) < \infty$ (with positive probability) are frequently referred to as limited-type policies. Similarly, disciplines with $\mathbf{L}_i(m_i) < \infty$ (with positive probability) may be viewed as decrementing-type policies.

In *binomial-type* policies, $\mathbf{K}_i(m_i) = \infty$ and $\mathbf{L}_i(m_i)$ is binomially distributed with mean $m_i q_i$, $0 \leq q_i \leq 1$, cf. Levy [138], Levy [139]. In *Bernoulli-type* policies, $\mathbf{L}_i(m_i) = m_i$ and $\mathbf{K}_i(m_i)$ is the sum of m_i independent identically geometrically distributed random variables each with mean $1/(1-q_i)$, $0 \leq q_i \leq 1$, cf. Resing [159]. In case $\mathbf{K}_i(m_i)$ is just a *single* geometrically distributed random variable, we obtain ordinary Bernoulli service, introduced in Keilson & Servi [124], [164]. Ordinary Bernoulli service may be used as an emulation of k_i-limited service, under which S serves at most k_i customers at Q_i (taking $q_i = 1 - 1/k_i$). Note that Bernoulli service and k_i-limited service coincide for $q_i = 0$ as well as $q_i = 1$. In its turn k_i-limited service is widely used as an approximation of time-limited service, under which the server stays at most for a time T_i at Q_i (taking $k_i \approx T_i/\beta_i$, the exact value depending on whether or not service is preempted when the timer expires). For deterministic service times k_i-limited service and T_i-limited service even coincide. A similar discipline is *fixed time* service under which the server stays at a queue for a fixed time, regardless of whether the queue becomes empty in the meanwhile or not. A service discipline under which S always serves a *fixed number* of customers at a queue does not really make sense as (for static routing policies) it inherently

causes the system to be unstable.
All above policies are local in the sense that the service decisions are made at each of the queues in isolation after the start of the visit. Recently Boxma, Levy, & Yechiali [50] proposed the *globally* gated service discipline as a modeling approach to reservation mechanisms like in the cyclic-reservation multiple-access (CRMA) protocol. Under globally gated service, during a visit to Q_i, S serves only the customers present at the start of the most recent visit to Q_1. As a generalization of ordinary gated and globally gated service, Khamisy, Altman, & Sidi [126] analyzed the *synchronized* gated service discipline. Under synchronized gated service, during a visit to Q_i, S serves only the customers present at the start of the most recent visit to a 'master' queue $Q_{\pi(i)}$ with $\pi(i) \in \{1, \ldots, n\}$. Synchronized versions of other service disciplines than gated, or service disciplines with other gating epochs than at the start of the most recent visit to $Q_{\pi(i)}$, e.g. at the *completion* of the most recent visit to $Q_{\pi(i)}$, are also conceivable, cf. Bertsekas & Gallager [16], Lee & Sengupta [136], [137]. Although global in nature, even in the latter policies the service decisions depend on the state of the system through the marginal queue length only. Policies in which the service decisions may be based on the joint queue length have been considered in Hofri & Ross [119], Koole [131], Liu, Nain, & Towsley [145]. With regard to the pro's and con's of such sophisticated adaptive service policies similar remarks hold as with regard to dynamic versus static routing policies.

The order of service
While at a queue, the order of service prescribes in which order S should serve customers. While the routing policy and the service policy together dictate the global priorities, the service order determines the local priorities. In the sequel the order of service is always assumed to be First Come First Served (FCFS), i.e., customers are assumed to be served in order of arrival. In fact the service order does neither matter for the queue length distribution nor, by Little's law, for the mean waiting times, as long as customers enter service in an order independent of their service times. Of course the service order *does* matter for the waiting-time *distribution*. Polling models with local priority rules within queues have been considered in Fournier & Rosberg [94], Shimogawa & Takahashi [165].

The stability condition
Finally, we briefly discuss the conditions for stability. Recently Fricker & Jaïbi [97] rigorously proved that for a system with periodic polling a necessary and sufficient condition for stability reads

$$\rho + \max_{i=1,\ldots,n} \lambda_i R/M_i < 1, \tag{1.1}$$

where R is the mean total switch-over time incurred during a cycle, i.e., incurred when passing through the polling table once, while M_i is the maximum mean number of type-i customers served during a cycle, i.e., the mean number of

type-i customers that would be served during a cycle if there were an infinite number of type-i customers present at the start of the cycle. Here the system is said to be stable if it admits a stationary regime with integrable cycle time. A simple traffic balance argument shows that if the system is stable then the server is working a fraction ρ of the time, so that the mean cycle time is given by $\mathbf{EC} = R/(1-\rho)$. So if the system is stable (1.1) may be rewritten as

$$\frac{\lambda_i R}{1-\rho} < M_i, \qquad i = 1, \ldots, n, \tag{1.2}$$

saying that the mean number of type-i customers arriving during a cycle is smaller than the maximum mean number of type-i customers served during a cycle.

As the server is working a fraction ρ_i of the time at Q_i, the mean *total* visit time at Q_i in a cycle is given by $\mathbf{EV}_i = \rho_i \mathbf{EC} = \rho_i R/(1-\rho)$, $i = 1, \ldots, n$. The mean *total* intervisit time at Q_i in a cycle follows from $\mathbf{EI}_i = \mathbf{EC} - \mathbf{EV}_i = (1-\rho_i)R/(1-\rho)$, $i = 1, \ldots, n$.

Note that if the system is said to be stable then all the queues are stable. However, even if a system is unstable a subset of the queues may still be stable. Assuming that the queues are indexed such that $i \leq j \iff \lambda_i/M_i \leq \lambda_j/M_j$, Fricker & Jaïbi show that the queues $Q_1, \ldots, Q_\kappa$ are stable while the queues $Q_{\kappa+1}, \ldots, Q_n$ are unstable with κ being defined as

$$\kappa := \max\{i : \sum_{j=1}^{i} \rho_j + \lambda_i (R + \sum_{j=i+1}^{n} \beta_j M_j)/M_i < 1\}.$$

In other words, whether or not the individual queues are stable depends on the ratio λ_i/M_i. The mean cycle time is given by $\mathbf{EC} = (R + \sum_{j=\kappa+1}^{n} \beta_j M_j)/(1 - \sum_{j=1}^{\kappa} \rho_j)$. Note that (1.1) implies $\kappa = n$.

The quantity M_i is determined by the number of visits to Q_i as specified in the polling table and the maximum mean number of customers served during a visit to Q_i as specified by the service discipline (possibly different policies at different visits). For ease of presentation we now focus on the case of strictly cyclic polling, so that $R = s$ and the quantity M_i is determined by the service discipline at Q_i only. (The sufficient stability conditions for the case of strictly cyclic polling were independently established by Altman, Konstantopoulos, & Liu [8] and Georgiadis & Szpankowski [108], using different techniques. The assumptions in [8] and [108] on the service disciplines are however somewhat restrictive compared to [97].) For service disciplines like exhaustive and gated that do not impose any (probabilistic) restriction on the maximum mean number of customers served, $M_i = \infty$, so that the stability condition (1.1) reduces to $\rho < 1$, which has long been stated without formal proof, cf. Eisenberg [84]. For both the exhaustive and gated version of k_i-limited service, $M_i = k_i$, so that (1.2) reduces to $\lambda_i s/(1-\rho) < k_i$, which also has long been stated without

formal proof, cf. Kühn [135]. For k_i-decrementing service, $M_i = k_i/(1-\rho_i)$, so that (1.2) reduces to $\lambda_i s(1-\rho_i)/(1-\rho) < k_i$, $i = 1, \ldots, n$, saying that the mean increase in the number of type-i customers during the intervisit time is smaller than the net decrease during the visit time.
In [98] Fricker & Jaïbi establish the stability condition for models with Markovian polling; cf. also Borovkov & Schassberger [25]. For dynamic scheduling strategies there are hardly any results known on the conditions for stability; cf. Schassberger [162] for the case of gated-limited service.
Throughout the thesis the conditions for stability are assumed to hold. Further we always assume the system under consideration to be in steady state.

1.4 Analysis of polling systems

In this section we survey the state of the art in the analysis of polling systems. Rather than covering all technical details, we intend to illuminate the main concepts in the analysis of polling systems, which contribute to putting the thesis in the right perspective. We refer to Takagi [172] for a thorough monograph on the analysis of polling systems, containing a detailed enumeration of the main results.

In one of the first polling studies, Avi-Itzhak, Maxwell, & Miller [14] study a two-queue model with zero switch-over times and alternating priority (i.e. exhaustive service at both queues). They obtain the sojourn time distribution by focusing on the system busy period. Takács [171] derives the waiting-time distribution in the same model by studying the Markov chain formed by the state of the system embedded at service completion epochs. Using similar techniques, Eisenberg [83] obtains the waiting-time distribution in a two-queue model with non-zero switch-over times and either alternating priority or strict priority, in which the server stops switching when the system is empty.
Cooper & Murray [77] study a model with an arbitrary number of queues, zero switch-over times, strictly cyclic polling, and exhaustive service at each of the queues. They derive the cycle time distribution by analyzing the Markov chain formed by the state of the system embedded at visit completion epochs. Cooper [78] obtains the waiting-time distribution for the model, by viewing the queues in isolation as vacation queues, the intervisit periods constituting the vacations. The solution method may also be used for a similar model with gated service at each of the queues.
Eisenberg [84] studies a model with an arbitrary number of queues, non-zero switch-over times, periodic polling, and exhaustive service at each of the queues, in which the server keeps switching when the system is empty. Eisenberg derives the waiting-time distribution, the marginal queue length distribution, and the joint queue length distribution at polling epochs by cleverly exploiting four Markov chains, embedded at service and visit beginnings and endings. The solution method may also be used for a similar model with gated service at

each of the queues. In a recent study [86], Eisenberg shows how an adapted version of the method may be applied in case the server stops switching when the system is empty.

Over the years several methods have been developed for computing the mean waiting times at the various queues in strictly cyclic polling systems with either exhaustive or gated service. To be specific, denote by $\mathbf{W}_i$ the waiting time of an arbitrary type-i customer, i.e, the time elapsing from its arrival to the start of its service, $i = 1, \ldots, n$.
One method for computing the mean waiting times is the *buffer occupancy* method as used by Cooper & Murray [77], Cooper [78], Eisenberg [84]. As the name suggests, in the buffer occupancy method the mean waiting times are computed starting from the *buffer occupancy* variables $\mathbf{X}_{ij}$, denoting the queue length at Q_j at the start of a visit to Q_i, $i, j = 1, \ldots, n$.
Define $F_i(z_1, \ldots, z_n) := \mathrm{E}(z_1^{\mathbf{X}_{i1}} \ldots z_n^{\mathbf{X}_{in}})$ to be the probability generating function (pgf) of the joint queue length distribution at the start of a visit to Q_i, $\mid z_j \mid \leq 1$, $j = 1, \ldots, n$. The buffer occupancy method starts with deriving n functional equations, expressing $F_{i+1}(\cdot)$ into $F_i(\cdot)$, $i = 1, \ldots, n$.
The waiting times are related to the buffer occupancy variables as follows, cf. Watson [187]. For exhaustive service, writing $\mathbf{X}_i$ for $\mathbf{X}_{ii}$, for $\mathrm{Re}\,\omega \geq 0$,

$$\mathrm{E}(\mathrm{e}^{-\omega \mathbf{W}_i}) = \frac{(1-\lambda_i\beta_i)\omega}{\omega - \lambda_i(1-\beta_i(\omega))} \frac{1 - \mathrm{E}((1-\omega/\lambda_i)^{\mathbf{X}_i})}{\omega \mathrm{E}\mathbf{X}_i/\lambda_i}, \tag{1.3}$$

the first term on the right-hand side representing the waiting-time Laplace-Stieltjes Transform (LST) in the corresponding isolated $M/G/1$ queue of Q_i with arrival rate λ_i and service time LST $\beta_i(\cdot)$.
For gated service, for $\mathrm{Re}\,\omega \geq 0$,

$$\mathrm{E}(\mathrm{e}^{-\omega \mathbf{W}_i}) = \frac{(1-\lambda_i\beta_i)\omega}{\omega - \lambda_i(1-\beta_i(\omega))} \frac{\mathrm{E}((\beta_i(\omega)^{\mathbf{X}_i}) - \mathrm{E}((1-\omega/\lambda_i)^{\mathbf{X}_i})}{(1-\lambda_i\beta_i)\omega \mathrm{E}\mathbf{X}_i/\lambda_i}, \tag{1.4}$$

the first term on the right-hand side standing again for the waiting-time LST in the corresponding isolated $M/G/1$ queue of Q_i. In Chapter 2 we will discuss the occurrence of that term in greater detail.
The above relationships yield expressions for $\mathrm{E}\mathbf{W}_i$ involving $f_i' := \mathrm{E}\mathbf{X}_i$ and $f_i'' := \mathrm{E}(\mathbf{X}_i(\mathbf{X}_i - 1)) = \mathrm{E}(\mathbf{X}_i^2) - \mathrm{E}\mathbf{X}_i$ as unknowns. There are explicit expressions available for the first moments $\mathrm{E}\mathbf{X}_i$; for exhaustive service $\mathrm{E}\mathbf{X}_i = \lambda_i \mathrm{E}\mathbf{I}_i = \lambda_i(1-\rho_i)s/(1-\rho)$; for gated service $\mathrm{E}\mathbf{X}_i = \lambda_i \mathrm{E}\mathbf{C} = \lambda_i s/(1-\rho)$. For the second moments $\mathrm{E}(\mathbf{X}_i^2)$ there are no explicit expressions available. However, the functional equations involving $F_i(\cdot)$, $i = 1, \ldots, n$, render a set of n^3 linear equations with n^3 unknowns $\mathrm{E}(\mathbf{X}_{ij}\mathbf{X}_{ik})$. The latter set of equations can best be solved in an iterative manner, which requires $O(n^3 \log_\rho \epsilon)$ elementary operations (additions, multiplications) with ϵ denoting the level of accuracy required.

Another method for computing the mean waiting times is the *station time* method as used in Ferguson & Aminetzah [93]. As the name reflects, in the station time method the mean waiting times are computed starting from the *station time* variables $\mathbf{U}_j$, denoting the length of the station time at Q_j, $j = 1, \ldots, n$. For exhaustive service the station time consists of the visit time plus the *preceding* switch-over time. For gated service the station time is composed of the visit time plus the *following* switch-over time.
Define $\Theta_i(\omega_1, \ldots, \omega_n) := \mathrm{E}(\mathrm{e}^{-\omega_1 \mathbf{U}_{i-1} - \ldots - \omega_n \mathbf{U}_{i-n}})$ (all indices mod n) to be the Laplace-Stieltjes Transform (LST) of the joint station time distribution of the last n visits at the start of a visit to Q_i, $\mathrm{Re}\,\omega_j \geq 0$, $j = 1, \ldots, n$. The station time method starts with establishing n functional equations expressing $\Theta_{i+1}(\cdot)$ into $\Theta_i(\cdot)$, $i = 1, \ldots, n$.
The waiting times are related to the station time variables as follows. For exhaustive service, for $\mathrm{Re}\,\omega \geq 0$,

$$\mathrm{E}(\mathrm{e}^{-\omega \mathbf{W}_i}) = \frac{(1 - \lambda_i \beta_i)\omega}{\omega - \lambda_i(1 - \beta_i(\omega))} \frac{1 - \mathrm{E}(\mathrm{e}^{-\omega \mathbf{I}_i})}{\omega \mathrm{E}\mathbf{I}_i}, \tag{1.5}$$

with $\mathbf{I}_i$ denoting the intervisit time at Q_i, i.e., $\mathbf{I}_i = \mathbf{S}_{i-1} + \mathbf{U}_{i-1} + \ldots + \mathbf{U}_{i-n+1}$. Note that (1.3) and (1.5) are equivalent by the fact that for exhaustive service $\mathbf{X}_i$ equals the number of arrivals at Q_i during $\mathbf{I}_i$, i.e., $\mathrm{E}(z^{\mathbf{X}_i}) = \mathrm{E}(\mathrm{e}^{-\lambda_i(1-z)\mathbf{I}_i})$.
For gated service, for $\mathrm{Re}\,\omega \geq 0$,

$$\mathrm{E}(\mathrm{e}^{-\omega \mathbf{W}_i}) = \frac{(1 - \lambda_i \beta_i)\omega}{\omega - \lambda_i(1 - \beta_i(\omega))} \frac{\mathrm{E}(\mathrm{e}^{-\lambda_i(1-\beta_i(\omega))\mathbf{C}_i}) - \mathrm{E}(\mathrm{e}^{-\omega \mathbf{C}_i})}{(1 - \lambda_i \beta_i)\omega \mathrm{E}\mathbf{C}_i}, \tag{1.6}$$

with $\mathbf{C}_i$ denoting the cycle time at Q_i, i.e., $\mathbf{C}_i = \mathbf{U}_{i-1} + \ldots + \mathbf{U}_{i-n}$. The equivalence of (1.4) and (1.6) follows from the fact that for gated service $\mathbf{X}_i$ equals the number of arrivals at Q_i during $\mathbf{C}_i$, i.e., $\mathrm{E}(z^{\mathbf{X}_i}) = \mathrm{E}(\mathrm{e}^{-\lambda_i(1-z)\mathbf{C}_i})$.
The above relationships yield expressions for $\mathrm{E}\mathbf{W}_i$ involving $g_i = \mathrm{E}\mathbf{U}_i$ and $g_{ij} = \mathrm{E}(\mathbf{U}_i \mathbf{U}_j)$ (Q_i being visited *before* Q_j) as unknowns. There are explicit expressions available for the means g_i; for exhaustive service $g_i = s_{i-1} + \mathrm{E}\mathbf{V}_i = s_{i-1} + \rho_i s/(1-\rho)$; for gated service $g_i = \mathrm{E}\mathbf{V}_i + s_i = \rho_i s/(1-\rho) + s_i$. For the covariances g_{ij} there are no explicit expressions available. However, the functional equations involving $\Theta_i(\cdot)$, $i = 1, \ldots, n$, induce a set of n^2 linear equations with n^2 unknowns g_{ij}. The latter set of equations can be solved in an iterative manner, which requires $O(n^2 \log_\rho \epsilon)$ elementary operations with ϵ denoting the level of accuracy demanded. A further advantage of the station time method in comparison with the buffer occupancy method is that the structure of the set of linear equations involved is somewhat simpler. Without explicitly solving it, Ferguson & Aminetzah observe from the structure of the set of linear equations that the intensity-weighted sum $\sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i$ yields a relatively simple expression in comparison with the extremely complicated expressions for the individual mean waiting times themselves, cf. also Watson [187].

For exhaustive service,

$$\sum_{i=1}^{n} \rho_i \mathrm{EW}_i = \rho \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}}{2(1-\rho)} + \rho \frac{s^{(2)}}{2s} + \frac{s}{2(1-\rho)} \left[\rho^2 - \sum_{i=1}^{n} \rho_i^2 \right]. \tag{1.7}$$

For gated service,

$$\sum_{i=1}^{n} \rho_i \mathrm{EW}_i = \rho \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}}{2(1-\rho)} + \rho \frac{s^{(2)}}{2s} + \frac{s}{2(1-\rho)} \left[\rho^2 + \sum_{i=1}^{n} \rho_i^2 \right]. \tag{1.8}$$

These relationships for the mean waiting times are commonly referred to as *pseudo-conservation laws*. In Chapter 2 we will discuss the existence of these pseudo-conservation laws in greater detail. A disadvantage of the station time method is that, unlike the buffer occupancy method, it only appears to be applicable to polling systems with either exhaustive or gated service.

Sarkar & Zangwill [160] describe a refinement of the station time method. They express the n^2 unknowns g_{ij} into the n unknowns g_{ii}, and derive a set of n linear equations for the latter coefficients, which however appears to be less sparse.

Konheim & Levy [129] describe a modification of the buffer occupancy method. They propose to calculate $\mathrm{E}(\mathbf{X}_i^2)$ by the so-called descendant set approach, which allows the computation of the mean waiting time at a single queue in only $O(n \log_\rho \epsilon)$ elementary operations with ϵ the level of accuracy desired.

Concluding, although efficient numerical evaluation of the mean waiting times is non-trivial, polling systems with exhaustive or gated service *do* allow an exact analysis for generally distributed service times, generally distributed switch-over times, and an arbitrary number of queues. Polling systems with limited or decrementing service however do not allow an exact analysis, apart from some special cases like two-queue cases and completely symmetric cases. Eisenberg [85] studies a two-queue model with zero switch-over times and alternating service (i.e. 1-limited service at both queues), transforming the problem of finding the joint queue length distribution into the problem of solving a singular Fredholm integral equation. Cohen & Boxma [74] analyze the same model, translating the problem into a Riemann-Hilbert boundary value problem. Using similar techniques, Boxma [37] studies a symmetric two-queue model with non-zero switch-over times and 1-limited service at both queues. Boxma & Groenendijk [43] analyze an asymmetric two-queue model with non-zero switch-over times and 1-limited service at both queues by formulating a Riemann boundary value problem. Cohen [75] considers a two-queue model with zero switch-over times and 1-decrementing (semi-exhaustive) service at both queues. The solution of the specific boundary value problem as formulated in each of the latter studies typically requires an arsenal of most advanced techniques from complex function theory, usually rendering contour-integral expressions for the mean waiting times. For polling systems with k-limited or k-decrementing service and $n > 2$ queues only approximative results are available, apart from some mean-value

results for global performance measures like the cycle times or for the waiting times in a completely symmetric system (cf. Fuhrmann [100] for the special case of 1-limited service).
Summarizing, we observe a striking difference in complexity between on the one hand service disciplines, like exhaustive and gated, that can be analyzed exactly in a general setting by standard methods and on the other hand service disciplines, like limited and decrementing, that can only be analyzed exactly in special cases by most ingenious techniques.
The existence of such a sharp distinction is illuminated in Resing [159] and independently explained in Fuhrmann [99]. Both Resing and Fuhrmann consider the class of service disciplines that satisfy the following property:

Property 1.4.1
If there are k_i customers present at Q_i at the start of a visit, then during the course of the visit, each of these k_i customers will effectively be replaced in an i.i.d. manner by a random population having probability generating function (pgf) $h_i(z_1, \ldots, z_n)$, which may be any n-dimensional pgf.

By using a multi-type branching process approach, both Resing and Fuhrmann show that the class of service disciplines that satisfy the above property allows an exact analysis. The results of Resing and Fuhrmann suggest that service disciplines that violate Property 1.4.1 defy an exact analysis, except for some special cases, like two-queue cases and completely symmetric cases.
The key element in their exposition is that if the service disciplines in a polling system satisfy Property 1.4.1, it is possible to relate the pgf $G_i(z_1, \ldots, z_n) := \mathrm{E}(z_1^{\mathbf{Y}_{i1}} \ldots z_n^{\mathbf{Y}_{in}})$ of the joint queue length distribution at the *end* of a visit to Q_i to the pgf $F_i(z_1, \ldots, z_n) := \mathrm{E}(z_1^{\mathbf{X}_{i1}} \ldots z_n^{\mathbf{X}_{in}})$ of the joint queue length distribution at the *beginning* of a visit to Q_i by

$$G_i(z_1, \ldots, z_n) = F_i(z_1, \ldots, z_{i-1}, h_i(z_1, \ldots, z_n), z_{i+1}, \ldots, z_n). \tag{1.9}$$

Moreover, it is possible to relate $F_{i+1}(z_1, \ldots, z_n)$ to $G_i(z_1, \ldots, z_n)$ by

$$F_{i+1}(z_1, \ldots, z_n) = G_i(z_1, \ldots, z_n)\sigma_i(\sum_{j=1}^{n} \lambda_j(1 - z_j)), \tag{1.10}$$

irrespective of the service disciplines (ignoring here some subtleties in case the total switch-over time in a cycle is zero, cf. Section 3). Thus we obtain $2n$ equations for $2n$ functions, which may be combined to obtain a functional equation for one of the functions $F_i(\cdot)$ or $G_i(\cdot)$, which may then be solved by a standard iterative procedure. As we will show in Section 2.1 most of the relevant performance measures like marginal queue lengths at an arbitrary epoch and waiting times may directly be derived from $F_i(\cdot)$ and $G_i(\cdot)$. Note that the approach of Resing and Fuhrmann is closely related to the buffer occupancy method, which was outlined earlier in the present section. In fact, the station

time method, which was also sketched there, only appears to be applicable to a very restricted subclass of the service disciplines satisfying Property 1.4.1.
Assuming the service disciplines to satisfy Property 1.4.1, Resing shows that in case the total switch-over time in a cycle is non-zero the joint queue length process at the polling epochs of a fixed but arbitrary queue constitutes a multi-type branching process with immigration in each state. The particle types in the branching process correspond to the customer types in the polling model, the offspring in the branching process represents the customers arriving during the service times in the polling model, and the immigration in the branching process corresponds to the customers arriving during the switch-over times in the polling model. In case the total switch-over time in a cycle is zero Resing shows that the joint queue length process at the polling epochs of a fixed queue constitutes a multi-type branching process with immigration in state zero only. The immigration in the branching process then corresponds to the customers arriving in an empty system in the polling model. So the models with zero and non-zero switch-over times are closely related through a common offspring generation, the only difference originating from the immigration. In Chapter 3 we expose the relationship between models with zero and non-zero switch-over times in greater technical detail.
The exhaustive service discipline satisfies Property 1.4.1 with $h_i(z_1, \ldots, z_n) = \eta_i(\sum_{j \neq i} \lambda_j(1-z_j))$. Here $\eta_i(\cdot)$ is the LST of the busy-period distribution in an ordinary isolated $M/G/1$ queue with arrival rate λ_i and service time distribution $B_i(\cdot)$, satisfying the functional equation $\eta_i(\omega) = \beta_i(\omega + \lambda_i(1 - \eta_i(\omega)))$, cf. [73] p. 250. The gated service discipline satisfies Property 1.4.1 with $h_i(z_1, \ldots, z_n) = \beta_i(\sum_{j=1}^{n} \lambda_j(1 - z_j))$. Limited and decrementing service disciplines violate Property 1.4.1 and have indeed not yielded an exact analysis, except for some special cases, like two-queue cases and completely symmetric cases.
Models with server-position dependent arrival rates and customer branching (as the word suggests), cf. Remark 1.3.1 and Remark 1.3.2, satisfy Property 1.4.1 and may thus be analyzed exactly by using a multi-type branching process approach. There are also some service disciplines, like synchronized gated, that strictly speaking do not satisfy Property 1.4.1 but still allow an exact analysis. Most of these service disciplines, however, satisfy the following generalization of Property 1.4.1:

Property 1.4.2
If there are k_i customers present at Q_i at the beginning (or the end) of a visit to $Q_{\pi(i)}$, with $\pi(i) \in \{1, \ldots, n\}$, then during the course of the visit to Q_i, each of these k_i customers will effectively be replaced in an i.i.d. manner by a random population having pgf $h_i(z_1, \ldots, z_n)$, which may be any n-dimensional pgf.

In Chapter 9 we will introduce another generalization of Property 1.4.1 to explore the class of service disciplines that allow an exact analysis in the case of

multiple servers.
Here we outlined the analysis for a continuous-time model with strictly cyclic polling and single Poisson arrivals. Without seriously complicating the analysis, continuous-time may be replaced by discrete-time, strictly cyclic polling may be generalized to periodic polling or Markovian polling, and single Poisson arrivals may be generalized to batch Poisson arrivals.

1.5 Optimization of polling systems

In this section we review the state of the art in the optimization of polling systems. As indicated before, we seek to identify the main issues in the optimization of polling systems, rather than exhaustively discuss all the achievements made. For extensive surveys on optimization of polling systems we refer to Boxma [39] (static optimization) and Yechiali [191] (semi-dynamic optimization).

In most of the polling applications some degree of freedom exists in the design or control of the system in choosing the parameters of the scheduling discipline (visit order, visit lengths, service order). A major objective in studying polling systems is to develop sufficient understanding of how these parameters influence the operation of the system and how these parameters should thus be chosen so as to improve the performance of the system. Nevertheless, compared to the well-trodden area of the analysis of polling systems, the field of optimization of polling systems still remains relatively unexplored. Although for example many waiting-time approximations have been proposed for limited-type service policies, the problem of determining appropriate values for the service limits has hardly been addressed.
In the optimization of polling systems the problem formulation is typically to optimize some measure of the system performance over some class of feasible scheduling disciplines. So there are two factors that play a role, first, what is the performance measure to be optimized, second, what is the class of feasible scheduling disciplines.
Concerning the first factor, there is probably no generic comprehensive measure to evaluate the system performance. Efficiency and fairness are commonly viewed as important aspects of the system performance. Although it is somewhat unclear exactly how efficiency and fairness should be defined, it is widely believed that there is some trade-off between them. On the one hand, exhaustive service is considered to be efficient but not very fair, as a heavily-loaded queue may dominate the complete system. On the other hand, 1-limited service is considered to be fair, as each of the queues receives at most one service per visit, but not very efficient. Ideally, a performance measure should relate all important aspects of the system performance to measurable quantities like waiting times, queue lengths, and excess probabilities, and should indicate how all those aspects weigh against each other. In view of these considerations

$\sum_{i=1}^{n} c_i \lambda_i \mathbf{EW}_i$ (by Little's law equivalent to $\sum_{i=1}^{n} c_i \mathbf{EL}_i$, with $\mathbf{L}_i$ denoting the number of waiting type-i customers at an arbitrary epoch) is widely accepted as a reasonable measure of the system performance.

Concerning the second factor, usually the class of feasible scheduling disciplines consists of a family of strategies of a similar structure that differ by some (vectorial) parameter. We now successively discuss some optimization studies that focus on optimization of a routing vector (routing probabilities, or polling table), a service vector (service probabilities, or service limits), and a routing vector and a service vector simultaneously. In addition we mention some optimization studies that analyze less structured dynamic polling policies.

Optimization of the routing policy for a given service policy

A considerable amount of research effort has been devoted to static optimization, i.e., optimization of static routing policies in which the routing decisions are made independently of the state of the system. Boxma, Levy, & Weststrate [47] (cf. also [188]) consider a system with random polling and at each of the queues either exhaustive or gated service. They address the problem of finding the routing probabilities $(p_1, \ldots, p_n)$ that minimize $\sum_{i=1}^{n} \rho_i \mathbf{EW}_i$, the latter quantity being explicitly known from the pseudo-conservation law for random polling, cf. [53]. Boxma, Levy, & Weststrate [39] (cf. also [188]) consider a system with periodic polling and at each of the queues either exhaustive, gated, or 1-limited service. They study the problem of determining a polling table that minimizes $\sum_{i=1}^{n} \rho_i \mathbf{EW}_i$. The proposed approach is to use the optimal visit ratios in random polling (i.e. the routing probabilities $(p_1, \ldots, p_n)$) as indication for the optimal visit ratios in periodic polling (i.e. the occurrence ratios of the queues in the polling table) and then to use the Golden Ratio procedure as heuristic for spacing the visits within the polling table. Boxma, Levy, & Weststrate [49] (cf. also [188]) address the generalized problem of determining a polling table that minimizes $\sum_{i=1}^{n} c_i \lambda_i \mathbf{EW}_i$, the latter quantity now being approximated in terms of the occurrence ratios of the queues in the polling table. Kruskal [134] studies a similar problem with deterministic arrival, service, and switch-over processes. In all cases the optimal visit ratios are given by surprisingly simple square-root formulae.

Also, a considerable amount of research effort has been put in semi-dynamic optimization, i.e., optimization of semi-dynamic routing policies in which periodically the visit order for some future period is determined, based on some partial knowledge of the state of the system. Browne & Yechiali [59] consider a system with either exhaustive or gated service at each of the queues. They address the problem of finding the visit order, at the start of each cycle, that minimizes the expected duration of the new cycle, based on knowledge of the queue lengths. The optimal visit order is given by a remarkably simple index-

type rule. However, they do not explicitly indicate how minimization of the expected duration of each new cycle is supposed to contribute to optimizing the system performance, in particular minimizing the mean waiting times. Actually, the mean cycle time remains $s/(1-\rho)$, cf. [8], in other words, the server is doing nothing but deferring the service of customers to future cycles. Results of Fabian & Levy [90] suggest that among all semi-dynamic visit orders the order that minimizes (maximizes) the expected duration of each new cycle in fact yields the largest (smallest) mean waiting times. Purely dynamic optimization of the routing policy (for a given service policy) has hardly received any attention so far.

Optimization of the service policy for a given routing policy
Borst, Boxma, & Levy [35] consider a system with a k-limited service strategy at each of the queues. They address the problem of determining the vector of service limits $(k_1, \ldots, k_n)$ that minimizes $\sum_{i=1}^{n} c_i \lambda_i \mathrm{E}\mathbf{W}_i$. Chapter 6 of the present thesis is based on the results obtained in [35]. Blanc & Van der Mei [23] study a similar optimization problem in a system with a Bernoulli service strategy at each of the queues. Purely dynamic optimization of the service policy (for a given routing policy) has hardly received any attention so far.

Simultaneous optimization of routing policy and service policy
Borst, Boxma, Harink, & Huitema [34] consider a system operated with a fixed time polling (ftp) scheme. An ftp scheme specifies which queue should be visited at what time, i.e., it specifies not only the *order* of the visits, but also the *starting times* of the visits. They address the problem of constructing an ftp scheme that minimizes $\sum_{i=1}^{n} c_i \lambda_i \mathrm{E}\mathbf{W}_i$. Chapter 7 of the present thesis is based on the results obtained in [34].
Liu, Nain, & Towsley [145] consider a system with a dynamic polling policy, i.e., a collection of instructions for making decisions on whether the server S should serve (which customer), switch (to which queue), or idle, based on some partial knowledge of the state of the system. They attempt to identify policies that stochastically minimize the total amount of work and the total number of customers present in the system at an arbitrary epoch. They show that optimal policies are exhaustive and greedy, i.e., the server should neither idle nor switch when a queue is non-empty. In addition they show that in a symmetric system optimal policies are patient and belong to the class of Stochastically Largest Queue policies, i.e., the server should remain idling at the last visited queue when the system is empty and the server should never switch to a queue known to be stochastically smaller than another queue.
For a model with zero switch-over times the optimal (non-preemptive) polling policy is known to be given by the $c\mu$-rule, cf. Meilijson & Yechiali [151], Buyukkoc, Varaiya, & Walrand [63]. For a symmetric two-queue model with non-zero switch-over times Hofri & Ross [119] show that the policy that mini-

mizes the sum of discounted switch-over times and the holding cost is exhaustive service in a nonempty system and is of threshold type for switching from an empty queue to another. For an asymmetric two-queue model with switch-over *costs* rather than switch-over *times* Koole [131] shows that the policy that minimizes the sum of discounted switching cost and holding cost is *not* a threshold policy, but that the best threshold policy approaches the optimal policy very well.

Monotonicity issues

Closely related to optimization issues are questions of stochastic ordering or monotonicity of the various performance measures with regard to the inducing stochastic processes (arrival process, service process, switch-over process) or with regard to the scheduling discipline (routing policy, service policy, order of service).

For a fixed but arbitrary routing policy, Levy, Sidi, & Boxma [142] establish a hierarchy of dominance relations among work-conserving, non-idling service policies with respect to the total amount of work in the system at any time. Under fairly mild assumptions, they show that the closer a service policy approaches the standard exhaustive service policy, the higher the service policy reaches in the hierarchy, so that in particular the standard exhaustive service policy figures at the top of the hierarchy.

Altman, Konstantopoulos, & Liu [8] consider a system with strictly cyclic polling and at each of the queues either exhaustive-type or gated-type service. They show that the queue lengths at polling epochs, the visit times, the intervisit times, and the cycle times are stochastically increasing in the arrival rates, the service times, and the switch-over times.

Note that the above ordering results refer to global performance measures or performance measures that are defined at polling epochs. For detailed performance measures like waiting times or queue lengths at arbitrary epochs there are hardly any ordering results known. In view of [8] it might be conjectured that also the waiting time and the queue length at Q_i are stochastically increasing with regard to the arrival rates, the service times, and the switch-over times as well as the 'limitedness' of the service at Q_i, but most of the statements of such tendency have either been disproved by simple counterexamples or have lacked proof so far. As one of the scarce results, we establish in Chapter 5 a stochastic ordering relation for the waiting times in a globally gated polling system.

1.6 Overview of the thesis

We now give an overview of the main results presented in the remainder of the thesis.

In Chapter 2 we elaborate on the use of decomposition properties to analyze polling models, in particular discussing the existence of so-called pseudo-

conservation laws for the mean waiting times, cf. (1.7), (1.8). Broadening the scope somewhat, we also demonstrate the use of such decomposition properties to study a related model, namely, a queueing system with a customer collection mechanism.

In Chapter 3 we consider two different single-server polling systems: (i) a model with *zero* switch-over times, and (ii) a model with *non-zero* switch-over times, in which the server keeps cycling when the system is empty. For both models we relate the steady-state queue length distribution at a queue to the queue length distribution at visit beginning and visit completion instants at that queue. As a by-product we obtain a shorter proof of the Fuhrmann-Cooper decomposition, cf. [103]. For the important class of service disciplines with a branching structure satisfying Property 1.4.1, we expose a strong relationship between both the queue length and the waiting-time distribution in the two models. We also show how the latter relationship can be exploited to reduce the computational complexity of numerical moment calculations.

In Chapters 4 and 5 we study a polling system with a *dormant* server, i.e., a polling system in which the server may be allowed to make a halt at a queue when there are no customers present in the system. In the polling literature the server is usually assumed never to idle, in other words, to be switching when not working. In particular the server is assumed to be switching when there are no customers present in the system. However, quite often there are very sound reasons for letting the server stop switching when there are no customers present in the system, rather than letting the server needlessly circle around. In Chapter 4 we therefore derive a pseudo-conservation law for a general model, permitting a variety of service disciplines, in which the server may be allowed to make a halt at an arbitrary subset of queues. We use the pseudo-conservation law to compare the dormant and the non-dormant server case. Further we address the question at which queues the server should make a halt to minimize the mean total amount of work in the system.

The option of idling especially goes hand in hand quite naturally with the globally gated service discipline, introduced in Boxma, Levy, & Yechiali [50]. Under the globally gated service, during a tour along the queues exactly those customers are served that were already present at the start of the tour, while the service of customers that meanwhile arrive in the system is deferred until the next tour, cf. Section 1.3. Thus, as suggested in Boxma, Weststrate, & Yechiali [54], it does not make sense to start a tour along the queues when there are no customers present in the system. In Chapter 5 we therefore focus on a globally gated polling system in which the server makes a halt at its home base when there are no customers present in the system. We derive an explicit expression for the LST of the cycle time distribution, for the LST of the waiting-time distribution at each of the queues, and for the pgf of the joint queue length distribution at polling epochs. As a justification of the dormant server policy, the waiting time at each of the queues is shown to be smaller (in the increasing-convex-ordering sense) than in the ordinary non-dormant server case.

In Chapters 6 and 7 we discuss several optimization issues in polling systems. In Chapter 6 we consider a polling system with a k-limited service strategy. Under k-limited service, when visiting a queue, the server works until either a prespecified number of customers have been served, or the queue becomes empty, whichever occurs first, cf. Section 1.3. We are interested in the problem of determining appropriate values for the service limits that contribute to an efficient operation of the system. It appears that if we do not impose any constraint on the k_i's, then at least one of the optimal k_i's is always infinite. To accomplish a bound on the cycle time we therefore also study a version of the problem with a constraint of the form $\sum_{i=1}^{n} \gamma_i k_i \leq K$. We propose four different approaches to the constrained optimization problem, based on four different approximations for the mean waiting times, which are extensively investigated by numerical experiments. Next we discuss some properties of polling systems with k-limited service and establish a (partially conjectured) $c\mu$-like rule for the unconstrained optimization problem. We then propose an approximative approach to this problem, which is also elaborately examined by numerical experiments.

In Chapter 7 we consider a polling system operated with a fixed time polling (ftp) scheme. An ftp scheme specifies which queue should be visited at what time, i.e., it specifies not only the *order* of the visits, but also the *starting times* of the visits. We are interested in the problem of constructing ftp schemes that contribute to an efficient operation of the system. Starting from rather simple approximations, we formulate the problem as a mathematical program. In view of its NP-hardness we develop a heuristic method for solving the mathematical program. The method is tested by numerical experiments.

In Chapters 8, 9, and 10 we consider multiple-queue systems with multiple servers. In Chapter 8 we consider a system consisting of several customer types attended by several parallel non-identical servers. Customers are allocated to the servers in a probabilistic manner; upon arrival customers are sent to one of the servers according to a matrix of routing probabilities. We consider the problem of finding an allocation that minimizes a weighted sum of the mean waiting times. We expose the structure of an optimal allocation and describe in detail for some special cases how the structure may be exploited in actually determining an optimal allocation. Further we consider the problem of finding an optimal deterministic allocation, i.e., an optimal allocation that involves a 0-1 matrix of routing probabilities. We show the problem to be NP-hard and indicate how the structure of an optimal non-deterministic allocation may be used as a heuristic guideline in searching for an optimal deterministic allocation.

In Chapters 9 and 10 we consider polling systems with multiple servers, in which the cooperation, unlike in the situation in Chapter 8, also results in actual interaction of the servers. So far there are hardly any exact results known for such multiple-server polling systems, apart from some mean-value results for global performance measures like cycle times. In Chapter 9 we consider

systems in which the servers are assumed to be *coupled*, i.e., the servers always visit the queues together. Guided by Property 1.4.1, we explore the class of systems that allow an exact analysis. For these systems we present distributional results for the waiting time, the marginal queue length, and the joint queue length at polling epochs. The class in question includes several single-queue systems with a varying number of servers, two-queue two-server systems with exhaustive service and exponential service times, as well as infinite-server systems with an arbitrary number of queues, exhaustive or gated service, and deterministic service times.
In Chapter 10 we consider systems in which the servers are assumed to be *independent*, i.e., each of the servers visits the queues according to its own cyclic schedule. These systems appear to completely defy the derivation of exact analytical waiting-time results, which motivates the search for accurate approximations. We derive such waiting-time approximations for systems with the exhaustive and gated service discipline. The approximations are tested for a wide range of parameter combinations.

The results presented in the next chapters were obtained during the course of the author's thesis research at CWI, Amsterdam. Some of the results have already appeared in the open literature. The results in Section 2.4 are based on Borst, Boxma, & Combé [32], [33]. In Chapter 3 we present the results obtained in Borst & Boxma [31]. The results of Chapter 4 are based on Borst [26], [27] and those of Chapter 5 on Borst [27], [28]. Chapter 6 is based on the results in Borst, Boxma, & Levy [35]. In Chapter 7 we present the results obtained in Borst, Boxma, Harink, & Huitema [34]. The results of Chapter 8 are based on Borst [30], and those of Chapter 9 on Borst [29]. Chapter 10 is based on the results obtained in Borst & Van der Mei [36] and Van der Mei & Borst [150].

Throughout the thesis, stochastic variables are denoted by capitals and printed bold. References to the literature are presented as the name(s) of the author(s) followed by the corresponding index in the reference list or, in case of repeated occurrence, as the index in the reference list only, omitting the name(s) of the author(s). The chapters are each divided in a number of sections. Formulas are numbered per chapter, e.g., formula (2.6) is the sixth formula in Chapter 2. Assumptions, corollaries, examples, figures, lemma's, properties, remarks, tables, theorems, etc. are numbered per section, e.g., Theorem 4.3.1 is the first theorem in Section 3 of Chapter 4.

Chapter 2

Decomposition properties and pseudo-conservation laws in polling models

2.1 Introduction

Overviewing the recent polling literature, we see a prominent role played by so-called pseudo-conservation laws, which provide exact expressions for a specific weighted sum of the mean waiting times, mostly $\sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i$, cf. Ferguson & Aminetzah [93], Watson [187]. Although the individual mean waiting times usually involve extremely complicated expressions, a pseudo-conservation law typically provides a relatively simple explicit expression, which e.g. depends on the switch-over times only through the first two moments of their sum. There are even service disciplines, like Bernoulli service, for which the individual mean waiting times are completely unknown, but for which a pseudo-conservation law is still explicitly known. Thus pseudo-conservation laws provide a useful measure of the overall system performance. In addition, pseudo-conservation laws prove to be a valuable instrument for constructing and validating waiting-time approximations, and for determining the mean waiting times in a completely symmetric system in a simple manner. Pseudo-conservation-law-based waiting-time approximations for models with 1-limited service and an arbitrary number of queues are developed e.g. in Boxma & Meister [51], [52], Groenendijk [113], Srinivasan [168]. Waiting-time approximations for similar models with k-limited service are presented e.g. in Chang & Sandhu [66], Everitt [87], [89], Fuhrmann & Wang [104].

The existence of pseudo-conservation laws is interpreted in Boxma & Groenendijk [42] and further clarified in Boxma [38]. The framework presented by Boxma & Groenendijk unified and generalized the pseudo-conservation laws ex-

isting until then and also explained why they existed. In addition, the approach allows a relatively simple derivation of pseudo-conservation laws and a probabilistic interpretation of the various terms occurring in pseudo-conservation laws. Until then, pseudo-conservation laws were obtained by cumbersome ad hoc methods which did not really explain why they existed and what the various terms occurring represented.

The key element in the framework presented by Boxma & Groenendijk is the property of *work decomposition* which builds on the fundamental property of *work conservation*, and is closely related to the concept of *queue length decomposition* as described in Fuhrmann & Cooper [103]. To illuminate these concepts, we consider in the present chapter a single-server queueing system, not necessarily a polling model, with a Poisson arrival process of rate λ, a service time distribution $B(\cdot)$ with LST $\beta(\cdot)$ and mean β, and *service interruptions*. The service interruptions are assumed to result from some kind of interfering process that from time to time may keep the server from working, even when there are customers present. A period during which the server is not working, because of a service interruption, or because there are no customers present, will be referred to as a non-serving interval.

The service interruptions may be interwoven with the arrival and service process in an arbitrarily complex manner, but may not anticipate on the arrival and service times of future customers. In particular the durations of successive service interruptions are allowed to be dependent.

For now we abstract from what kind of interfering process causes the service interruptions. In a performability setting a service interruption typically represents a down-period of the system. In the context of polling models a service interruption usually corresponds to a switch-over time, or to an intervisit period with regard to a specific queue, depending upon whether the polling system in totality is viewed as a system with service interruptions, or just a specific queue in isolation. Broadening the scope, the service interruptions may also represent set-up times, shut-down times, reconfiguration times, periods during which the server performs some secondary tasks, or they may correspond to some collection or reservation mechanism for customers on which we will focus in Section 2.4.

The remainder of the chapter is organized as follows. In Section 2.2 we describe the property of queue length decomposition for the model under consideration. Focusing on a specific queue in a polling system in isolation, we indicate how the queue length decomposition also translates into a decomposition of the waiting time, as already recognized in Section 1.4, cf. (1.3), (1.4), (1.5), (1.6). In Section 2.3 we elaborate on the related property of work decomposition. Applying it to a polling system in totality, we sketch how the work decomposition property leads to pseudo-conservation laws for the mean waiting times, as already observed in Section 1.4, cf. (1.7), (1.8). In Section 2.4 we demonstrate how the queue length and work decomposition properties may be exploited in a related model, namely, a queueing system with a customer collection mechanism.

2.2 QUEUE LENGTH DECOMPOSITION

Using concepts from the theory of branching processes, under rather mild assumptions, Fuhrmann & Cooper [103] prove the following *queue length decomposition* property for the model under consideration:

$$\mathbf{N} \stackrel{d}{=} \mathbf{N}_{M/G/1} + \mathbf{N}_I, \tag{2.1}$$

with $\stackrel{d}{=}$ denoting equality in distribution;
$\mathbf{N}$:= the queue length at an arbitrary epoch;
$\mathbf{N}_{M/G/1}$:= the queue length at an arbitrary epoch in the 'corresponding' $M/G/1$ system;
$\mathbf{N}_I$:= the queue length at an arbitrary epoch in a non-serving interval;
$\mathbf{N}_{M/G/1}$ and $\mathbf{N}_I$ being independent.
The corresponding $M/G/1$ system is an ordinary $M/G/1$ queue with similar traffic characteristics, but without any service interruptions. To find the distribution of $\mathbf{N}$, it thus suffices to find the distribution of $\mathbf{N}_I$, as the distribution of $\mathbf{N}_{M/G/1}$ is simply known from the Pollaczek-Khintchine formula, cf. [73] p. 238. From a methodological point of view however it is usually preferable to analyze the queue length at the beginning and the end of a non-serving interval rather than to study $\mathbf{N}_I$, the queue length at an arbitrary epoch in a non-serving interval. Therefore we now relate the distribution of $\mathbf{N}_I$ to the queue length distribution at such embedded epochs. Denote by $\mathbf{N}_{\text{begin}}^{(k)}$ and $\mathbf{N}_{\text{end}}^{(k)}$ the queue length at the beginning and the end of the k-th non-serving interval. Denote by $\mathbf{N}_{\text{begin}}$ and $\mathbf{N}_{\text{end}}$ a pair of stochastic variables with as joint distribution the stationary joint distribution of $\mathbf{N}_{\text{begin}}^{(k)}$ and $\mathbf{N}_{\text{end}}^{(k)}$.

Lemma 2.2.1

$$\Pr\{\mathbf{N}_I = l\} = \frac{\Pr\{\mathbf{N}_{\text{begin}} \leq l\} - \Pr\{\mathbf{N}_{\text{end}} \leq l\}}{\mathrm{E}\mathbf{N}_{\text{end}} - \mathrm{E}\mathbf{N}_{\text{begin}}}. \tag{2.2}$$

Written in terms of pgf's,

$$\mathrm{E}(z^{\mathbf{N}_I}) = \frac{\mathrm{E}(z^{\mathbf{N}_{\text{begin}}}) - \mathrm{E}(z^{\mathbf{N}_{\text{end}}})}{(1-z)(\mathrm{E}\mathbf{N}_{\text{end}} - \mathrm{E}\mathbf{N}_{\text{begin}})}, \qquad |z| \leq 1. \tag{2.3}$$

Proof
Because of the PASTA property $\mathbf{N}_I$ has the same distribution as the number of customers seen by an arbitrary customer arriving in a non-serving interval. In the first K non-serving intervals, the fraction of customers that see l customers upon arrival is

$$\frac{\sum_{k=1}^{K} \mathrm{I}_{\{\mathbf{N}_{\text{begin}}^{(k)} \leq l \leq \mathbf{N}_{\text{end}}^{(k)} - 1\}}}{\sum_{k=1}^{K} (\mathbf{N}_{\text{end}}^{(k)} - \mathbf{N}_{\text{begin}}^{(k)})},$$

with $I_{\{E\}}$ denoting the indicator function of the event E. So, using the law of large numbers,

$$\begin{aligned}
\Pr\{\mathbf{N}_I = l\} &= \lim_{K\to\infty} \frac{\sum_{k=1}^{K} I_{\{\mathbf{N}_{\text{begin}}^{(k)} \le l \le \mathbf{N}_{\text{end}}^{(k)} - 1\}}/K}{\sum_{k=1}^{K} (\mathbf{N}_{\text{end}}^{(k)} - \mathbf{N}_{\text{begin}}^{(k)})/K} \\
&= \frac{\Pr\{\mathbf{N}_{\text{begin}} \le l \le \mathbf{N}_{\text{end}} - 1\}}{\mathrm{E}\mathbf{N}_{\text{end}} - \mathrm{E}\mathbf{N}_{\text{begin}}} \\
&= \frac{\Pr\{\mathbf{N}_{\text{begin}} \le l\} - \Pr\{\mathbf{N}_{\text{end}} \le l\}}{\mathrm{E}\mathbf{N}_{\text{end}} - \mathrm{E}\mathbf{N}_{\text{begin}}},
\end{aligned}$$

as $\Pr\{\mathbf{N}_{\text{begin}} \le \mathbf{N}_{\text{end}}\} = 1$.

□

The queue length decomposition property (2.1) only holds if the order of service is independent of the service times. Consequently, it typically does *not* hold for a polling system in totality, but it *does* hold for each of the queues in isolation, assuming that the order of service at each of the queues is independent of the service times. Therefore, we now focus on the queue length at a specific queue in a polling system in isolation, let us say Q_i. For the specification of the arrival, service, and switch-over processes we refer to the description of the 'basic model' in Section 1.3. (In Section 2.4 we use the queue length decomposition property to analyze a system where the service interruptions arise from a customer collection procedure.) The queue length $\mathbf{N}_{\text{begin}}$ and $\mathbf{N}_{\text{end}}$ at the *beginning* and the *end* of a *non-serving interval* then correspond to the queue length $\mathbf{Y}_i$ at the *end* and $\mathbf{X}_i$ at the *beginning* of a *visit* to Q_i, respectively. Applying (2.1) and (2.3) to Q_i we then obtain

$$\mathbf{N}_i \stackrel{d}{=} \mathbf{N}_{i|M/G/1} + \mathbf{N}_{i|I}, \tag{2.4}$$

$$\mathrm{E}(z^{\mathbf{N}_{i|I}}) = \frac{\mathrm{E}(z^{\mathbf{Y}_i}) - \mathrm{E}(z^{\mathbf{X}_i})}{(1-z)(\mathrm{E}\mathbf{X}_i - \mathrm{E}\mathbf{Y}_i)}, \qquad |z| \le 1, \tag{2.5}$$

with

$\mathbf{N}_i :=$ the queue length at Q_i at an arbitrary epoch;

$\mathbf{N}_{i|M/G/1} :=$ the queue length at an arbitrary epoch in the 'corresponding' $M/G/1$ queue of Q_i in isolation;

$\mathbf{N}_{i|I} :=$ the queue length at Q_i at an arbitrary epoch in an intervisit period;

$\mathbf{N}_{i|M/G/1}$ and $\mathbf{N}_{i|I}$ being independent.

By (2.4) and (2.5), to find the queue length distribution at Q_i at an arbitrary epoch it suffices to find the queue length distribution at Q_i at the beginning and the end of a visit, respectively. Note that for the class of service disciplines that satisfy Property 1.4.1, $\mathrm{E}(z^{\mathbf{X}_i}) = F_i(1,\ldots,1,z,1,\ldots,1)$ and $\mathrm{E}(z^{\mathbf{Y}_i}) = G_i(1,\ldots,1,z,1,\ldots,1)$, with z as i-th argument, are actually known.

We now abstract from the polling context again. If the order of service in the system with service interruptions is FCFS, then the sojourn time $\mathbf{R}$ of an arbitrary customer, i.e., the time from its arrival to the completion of its service, is related to the queue length $\mathbf{N}$ (the total number of customers present, including a customer possibly in service) at an arbitrary epoch by the distributional form of Little's law, cf. Keilson & Servi [125]:

$$\mathrm{E}(z^{\mathbf{N}}) = \mathrm{E}(\mathrm{e}^{-\lambda(1-z)\mathbf{R}}), \qquad |\, z \,| \leq 1, \tag{2.6}$$

provided the sojourn time of customers is independent of arrivals after their own arrival.
Similarly, the waiting time $\mathbf{W}$ of an arbitrary customer is related to the number $\mathbf{L}$ of waiting customers (excluding a customer possibly in service):

$$\mathrm{E}(z^{\mathbf{L}}) = \mathrm{E}(\mathrm{e}^{-\lambda(1-z)\mathbf{W}}), \qquad |\, z \,| \leq 1. \tag{2.7}$$

Taking expectations in either (2.6) or (2.7) yields the original form of Little's law. Combining (2.1) and (2.6), taking $\lambda(1-z) = \omega$, we obtain the following sojourn time decomposition:

$$\mathrm{E}(\mathrm{e}^{-\omega\mathbf{R}}) = \mathrm{E}(\mathrm{e}^{-\omega\mathbf{R}_{M/G/1}})\mathrm{E}((1-\omega/\lambda)^{\mathbf{N}_I}), \qquad \mathrm{Re}\,\omega \geq 0, \tag{2.8}$$

with $\mathbf{R}_{M/G/1}$ denoting the sojourn time of an arbitrary customer in the corresponding $M/G/1$ system.
Noting that $\mathrm{E}(\mathrm{e}^{-\omega\mathbf{R}}) = \mathrm{E}(\mathrm{e}^{-\omega\mathbf{W}})\beta(\omega)$, we obtain from (2.8) the following waiting-time decomposition:

$$\mathrm{E}(\mathrm{e}^{-\omega\mathbf{W}}) = \mathrm{E}(\mathrm{e}^{-\omega\mathbf{W}_{M/G/1}})\mathrm{E}((1-\omega/\lambda)^{\mathbf{N}_I}), \qquad \mathrm{Re}\,\omega \geq 0, \tag{2.9}$$

with $\mathbf{W}_{M/G/1}$ denoting the waiting time of an arbitrary customer in the corresponding $M/G/1$ system.
We now return to the polling setting again. As the order of service is required to be FCFS here, the properties (2.6)-(2.9) typically do *not* hold for a polling system in totality, but they *do* hold for a specific queue in isolation with FCFS order of service. Applying (2.9) to Q_i, using (2.5) and the Pollaczek-Khintchine formula, cf. [73] p. 255, we obtain

$$\mathrm{E}(\mathrm{e}^{-\omega\mathbf{W}_i}) = \frac{(1-\lambda_i\beta_i)\omega}{\omega - \lambda_i(1-\beta_i(\omega))} \frac{\mathrm{E}((1-\omega/\lambda_i)^{\mathbf{Y}_i}) - \mathrm{E}((1-\omega/\lambda_i)^{\mathbf{X}_i})}{(\mathrm{E}\mathbf{X}_i - \mathrm{E}\mathbf{Y}_i)\omega/\lambda_i}.$$

Taking for exhaustive service $\mathbf{Y}_i \equiv 0$ and $\mathrm{E}(z^{\mathbf{X}_i}) = \mathrm{E}(\mathrm{e}^{-\lambda_i(1-z)\mathbf{I}_i})$ leads to (1.3) and (1.5). Noting that for gated service $\mathrm{E}(z^{\mathbf{Y}_i}) = \mathrm{E}(\beta_i(\lambda_i(1-z))^{\mathbf{X}_i})$ and $\mathrm{E}(z^{\mathbf{X}_i}) = \mathrm{E}(\mathrm{e}^{-\lambda_i(1-z)\mathbf{C}_i})$ gives (1.4) and (1.6).

2.3 Work decomposition

In the previous section we described the property of *queue length decomposition* and indicated how it also translates into a decomposition of the waiting times at the individual queues in a polling system. In this section we focus on the related property of *work decomposition* and sketch how it leads to pseudo-conservation laws for the mean waiting times in a polling system, as shown by Boxma & Groenendijk [42].

Adapting the arguments of Fuhrmann & Cooper, under even milder assumptions, Boxma & Groenendijk prove the following *work decomposition* property for the model under consideration:

$$\mathbf{V} \stackrel{d}{=} \mathbf{V}_{M/G/1} + \mathbf{V}_I, \tag{2.10}$$

with

$\mathbf{V}$:= the amount of work in the system at an arbitrary epoch;

$\mathbf{V}_{M/G/1}$:= the amount of work at an arbitrary epoch in the corresponding $M/G/1$ system;

$\mathbf{V}_I$:= the amount of work in the system at an arbitrary epoch in a non-serving interval;

$\mathbf{V}_{M/G/1}$ and $\mathbf{V}_I$ being independent.

When the amount of work in a non-serving interval were always zero, i.e., $\mathbf{V}_I \equiv 0$, (2.10) would reduce to the fundamental property of *work conservation*, which in fact holds even in sample-path sense. Note that in a polling system in general it is not the case that $\mathbf{V}_I \equiv 0$, as the server may be switching when there are customers present.

The work decomposition property still holds if the order of service is not independent of the service times, reflecting that in a sense the amount of work in the system is a less sensitive quantity than the queue length. Consequently, it holds in particular for the total amount of work in a polling system. Therefore, we now focus on the total amount of work in a polling system, and show how the work decomposition property leads to a pseudo-conservation law for the mean waiting times. For the specification of the arrival, service, and switch-over processes we refer to the description of the 'basic model' in Section 1.3. (In Section 2.4 we use the work decomposition property to analyze a system where the service interruptions originate from a customer collection procedure.)

Applying Brumelle's formula [60],

$$\mathrm{E}\mathbf{V} = \sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i + \frac{1}{2} \sum_{i=1}^{n} \lambda_i \beta_i^{(2)}. \tag{2.11}$$

From the Pollaczek-Khintchine formula, cf. [73] p. 255,

$$\mathrm{E}\mathbf{V}_{M/G/1} = \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}}{2(1-\rho)}. \tag{2.12}$$

Taking expectations in (2.10), substituting (2.11), (2.12), we obtain the following relationship for the mean waiting times:

$$\sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i = \rho \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}}{2(1-\rho)} + \mathrm{E}\mathbf{V}_I. \tag{2.13}$$

When the amount of work in a non-serving interval were always zero, i.e., $\mathrm{E}\mathbf{V}_I = 0$, (2.13) would reduce to the property that $\sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i$ does not depend on the scheduling discipline as long as $\mathrm{E}\mathbf{V}_I = 0$, which is commonly referred to as a *conservation law.*

For strictly cyclic polling Boxma & Groenendijk [42] show that $\mathrm{E}\mathbf{V}_I$ may be determined as follows:

$$\mathrm{E}\mathbf{V}_I = \rho \frac{s^{(2)}}{2s} + \frac{s}{2(1-\rho)} \left[\rho^2 - \sum_{i=1}^{n} \rho_i^2 \right] + \sum_{i=1}^{n} \mathrm{E}\mathbf{Z}_{ii}, \tag{2.14}$$

with $\mathbf{Z}_{ii}$ denoting the amount of work left behind by the server at Q_i at the completion of a visit.

Substituting (2.14) into (2.13), we obtain the following relationship for the mean waiting times:

$$\sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i = \rho \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}}{2(1-\rho)} + \rho \frac{s^{(2)}}{2s} + \frac{s}{2(1-\rho)} \left[\rho^2 - \sum_{i=1}^{n} \rho_i^2 \right] + \sum_{i=1}^{n} \mathrm{E}\mathbf{Z}_{ii}, \tag{2.15}$$

which is commonly referred to as *pseudo-conservation law*, as $\sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i$ now *does* depend on the scheduling discipline through the terms $\mathrm{E}\mathbf{Z}_{ii}$. As a pleasing circumstance however, $\mathrm{E}\mathbf{Z}_{ii}$ is determined by the service discipline at Q_i only, i.e., not by the service discipline at Q_j, $j \neq i$, e.g. [42]:

I. Exhaustive service:

$$\mathrm{E}\mathbf{Z}_{ii} = 0.$$

II. Gated service:

$$\mathrm{E}\mathbf{Z}_{ii} = \frac{\rho_i^2 s}{1-\rho}.$$

III. 1-Limited service:

$$\mathrm{E}\mathbf{Z}_{ii} = \frac{\rho_i^2 s}{1-\rho} + \rho_i \frac{\lambda_i s}{1-\rho} \mathrm{E}\mathbf{W}_i.$$

IV. 1-Decrementing service:

$$\mathrm{E}\mathbf{Z}_{ii} = -\frac{\rho_i^2 \lambda_i^2 \beta_i^{(2)} s}{2(1-\rho)} + \rho_i \frac{\lambda_i (1-\rho_i) s}{1-\rho} \mathrm{E}\mathbf{W}_i.$$

Substituting the above expressions into (2.15) yields the pseudo-conservation laws which before were only known to hold in cases with the same service disciplines at each of the queues.
Here we sketched the derivation of a pseudo-conservation law for a continuous-time model with strictly cyclic polling and single Poisson arrivals. Without seriously complicating the derivation, continuous-time may be replaced by discrete-time, cf. [44], strictly cyclic polling may be generalized to periodic polling, cf. [45], or Markovian polling, cf. [53], and single Poisson arrivals may be generalized to batch Poisson arrivals, cf. [38]. In Chapter 4 we show how a pseudo-conservation law is derived in case the server may be allowed to make a halt at a queue when there are no customers present in the system.

2.4 A QUEUEING SYSTEM WITH A CUSTOMER COLLECTION MECHANISM

In the two previous sections we sketched how decomposition properties may be exploited in the setting of polling models. In this section we describe how such decomposition properties may be applied to a related model, namely, a queueing system with a customer collection mechanism. For a detailed analysis of an aggregated version of the model to be studied we refer to Borst, Boxma, & Combé [32], [33]. For a generalization and unification of the results to be presented we refer to Boxma & Combé [40], Combé [76].
The model under consideration consists of n queues, $Q_1, \ldots, Q_n$, attended by a single server S. For the specification of the arrival and service processes at the queues we refer to the description of the 'basic model' in Section 1.3. Unlike in a polling model, where the server periodically visits the queues to serve (some of) the customers present, here a *collector* from time to time visits the queues, picks up (some of) the customers present, and delivers them to the server where they are served. Collectors are assumed to be sent out according to a Poisson process of rate γ, independent of the arrival and service processes at the queues. A collector visits the queues in strictly cyclic order, $Q_1, \ldots, Q_n$. Moving from Q_i to Q_{i+1} requires a constant travel time σ_i, $i = 1, \ldots, n$. Here σ_n is to be understood as the travel time from Q_n to the server. (Note that the travel time from the origin of the collectors to Q_1 is irrelevant here.)
Upon arrival at Q_i, the collector instantaneously picks up all the type-i customers that have already been present for at least a constant time τ_i (e.g. $\tau_i = 0$), $i = 1, \ldots, n$. The collectors are assumed to have infinite capacity. Upon arrival at the server, the collector instantaneously delivers all the customers that were picked up. The order of service is assumed not to discriminate between the various customer types; within the various customer classes the order of service is assumed to be FCFS.
Conceptually, a batch of collected customers may also be viewed as a single 'super' customer. The service time of the super customer comprises the total service time of the corresponding batch. So a super customer has a zero service time in case a batch happens to be empty. The arrival epoch of the super

customer coincides with the arrival epoch of the corresponding collector at the server. As the collectors have a constant travel time, the super customers arrive at the server according to a Poisson process of rate γ. Focusing on the super customers, the model under consideration may thus be viewed as an ordinary $M/G/1$ queue, essentially differing however from the standard model description due to the intrinsic *dependence* between the interarrival and service times. Namely, the larger the interarrival time of a super customer, the larger the corresponding intercollector time, the larger the total service time of the corresponding batch, and the larger the service time of the super customer. To be specific, denote by $(\mathbf{A}, \mathbf{B})$ a pair of stochastic variables with as joint distribution the joint distribution of the interarrival and service time of an arbitrary super customer. From [32], [33],

$$\mathrm{E}(\mathrm{e}^{-\zeta\mathbf{A}-\omega\mathbf{B}}) = \frac{\gamma}{\gamma+\zeta+\lambda(1-\beta(\omega))}, \qquad \mathrm{Re}\,\zeta \geq 0, \mathrm{Re}\,\omega \geq 0, \tag{2.16}$$

with $\beta(\omega) := \sum_{i=1}^{n} \frac{\lambda_i}{\lambda}\beta_i(\omega)$, so that $\mathrm{Cov}(\mathbf{A}, \mathbf{B}) = \lambda\beta/\gamma^2$ with $\beta = \sum_{i=1}^{n} \frac{\lambda_i}{\lambda}\beta_i = \rho/\lambda$. In the sequel we will not really highlight the effects of the dependence between the interarrival and service times, but the impact is studied in detail in [32, 33, 40, 76].

For future convenience, we first observe that the collection procedure under consideration may be equivalently defined as follows. Arriving type-i customers are first 'retained' for a while and 'released' after a constant time $\tau_i + \sum_{j=i}^{n} \sigma_j$. Upon arrival at Q_i the collector instantaneously picks up only the *released* type-i customers. Moving from Q_i to Q_{i+1} requires no travel time.

Taking the perspective of the last paragraph we immediately see that at every epoch the population of released customers is independent of the population of customers being retained, implying

$$\mathbf{V} \stackrel{d}{=} \mathbf{V}^* + \mathbf{V}^\#, \tag{2.17}$$

with

$\mathbf{V}$:= the amount of work in the system at an arbitrary epoch;

$\mathbf{V}^*$:= the amount of work in the system at an arbitrary epoch corresponding to *released* customers;

$\mathbf{V}^\#$:= the remaining amount of work in the system at an arbitrary epoch corresponding to customers being retained;

$\mathbf{V}^*$ and $\mathbf{V}^\#$ being independent.

The distribution of $\mathbf{V}^\#$ is given by

$$\mathrm{E}(\mathrm{e}^{-\omega\mathbf{V}^\#}) = \exp[-\sum_{i=1}^{n} \lambda_i(1-\beta_i(\omega))(\tau_i + \sum_{j=i}^{n} \sigma_j)], \qquad \mathrm{Re}\,\omega \geq 0. \tag{2.18}$$

Denote by $\mathbf{V}^0$ the amount of work at an arbitrary epoch in the 'corresponding' system with a zero-delay collection procedure, i.e., a system with similar traffic

characteristics, but with $\sigma_i = 0$, $\tau_i = 0$, $i = 1, \ldots, n$.
Observing that the population of released customers in the original system evolves similarly as the total population of customers in the corresponding zero-delay system,

$$\mathbf{V}^* \stackrel{d}{=} \mathbf{V}^0. \tag{2.19}$$

As described in Section 2.1, the following decomposition property holds for $\mathbf{V}^0$, cf. [38]:

$$\mathbf{V}^0 \stackrel{d}{=} \mathbf{V}_{M/G/1} + \mathbf{V}_I^0, \tag{2.20}$$

with
$\mathbf{V}_{M/G/1}$:= the amount of work at an arbitrary epoch in the 'corresponding' $M/G/1$ system without customer collection mechanism, i.e., a system with similar traffic characteristics, where the customers have immediate access to the server;
$\mathbf{V}_I^0$:= the amount of work in the zero-delay system at an arbitrary epoch in a non-serving interval;
$\mathbf{V}_{M/G/1}$ and $\mathbf{V}_I^0$ being independent.
For future convenience, we first introduce some further terminology. A 'basic' non-serving interval is a non-serving interval beginning at the departure of a super customer that leaves no super customers behind and ending at the arrival of the next super customer (which sees no super customers present). As super customers arrive according to a Poisson process of rate γ, the length of a basic non-serving interval is exponentially distributed with mean $1/\gamma$. In case the next arriving super customer (which sees no super customers present) happens to have zero service time, a number of consecutive basic non-serving intervals may occur, together constituting a larger non-serving interval. A 'maximal' non-serving interval is a non-serving interval that is not strictly contained in any other larger non-serving interval. Note that the length of a maximal non-serving interval is no longer exponentially distributed.
The distribution of $\mathbf{V}_{M/G/1}$ in (2.20) follows from the Pollaczek-Khintchine formula, cf. [73] p. 255,

$$\mathrm{E}(\mathrm{e}^{-\omega \mathbf{V}_{M/G/1}}) = \frac{(1-\lambda\beta)\omega}{\omega - \lambda(1-\beta(\omega))}, \qquad \mathrm{Re}\,\omega \geq 0. \tag{2.21}$$

The quantity $\mathbf{V}_I^0$ in (2.20) may be equivalently defined as the amount of work in the zero-delay system at an arbitrary epoch in a basic non-serving interval. Let us say D is the super customer at whose departure the basic non-serving interval in question started.
The quantity $\mathbf{V}_I^0$ then consists of two independent components, viz.,

$$\mathbf{V}_I^0 \stackrel{d}{=} \mathbf{V}_I' + \mathbf{V}_I'', \tag{2.22}$$

with
$\mathbf{V}_I'$:= the amount of work that arrived since the departure of D;

$\mathbf{V}_I''$:= the amount of work that arrived during the sojourn time of D;
$\mathbf{V}_I'$ and $\mathbf{V}_I''$ being independent.
For any non-negative stochastic variable $\mathbf{T}$, denote by $\mathbf{V}(\mathbf{T})$ the amount of work arriving to the system during a period of length $\mathbf{T}$, i.e.,

$$\mathrm{E}(\mathrm{e}^{-\omega \mathbf{V}(\mathbf{T})}) = \mathrm{E}(\mathrm{e}^{-\lambda(1-\beta(\omega))\mathbf{T}}), \qquad \mathrm{Re}\,\omega \geq 0. \tag{2.23}$$

The quantity $\mathbf{V}_I'$ is the amount of work arriving during the past of the basic non-serving interval in question at the arrival of D. As a basic non-serving interval is exponentially distributed, also the past of a basic non-serving interval is exponentially distributed. So $\mathbf{V}_I'$ is distributed as the amount of work arriving during an arbitrary interarrival time $\mathbf{A}$, i.e., distributed as the service time $\mathbf{B}$ of an arbitrary super customer,

$$\mathbf{V}_I' \stackrel{d}{=} \mathbf{V}(\mathbf{A}) \stackrel{d}{=} \mathbf{B}, \tag{2.24}$$

with, taking $\zeta = 0$ in (2.16),

$$\mathrm{E}(\mathrm{e}^{-\omega \mathbf{B}}) = \frac{\gamma}{\gamma + \lambda(1-\beta(\omega))}, \qquad \mathrm{Re}\,\omega \geq 0. \tag{2.25}$$

Denote by $\tilde{\mathbf{R}}$ the sojourn time of D.

$$\mathbf{V}_I'' \stackrel{d}{=} \mathbf{V}(\tilde{\mathbf{R}}). \tag{2.26}$$

Substituting (2.19), (2.20), (2.22), (2.24), (2.26) into (2.17), we obtain the following detailed form of the work decomposition property:

$$\mathbf{V} \stackrel{d}{=} \mathbf{V}_{M/G/1} + \mathbf{V}(\mathbf{A}) + \mathbf{V}(\tilde{\mathbf{R}}) + \mathbf{V}^{\#}. \tag{2.27}$$

We now show how a functional equation may be derived for the LST $r(\omega) := \mathrm{E}(\mathrm{e}^{-\omega \mathbf{R}})$, $\mathrm{Re}\,\omega \geq 0$, of the sojourn time distribution of an arbitrary super customer.
Because of the PASTA property

$$\mathbf{V}^* \stackrel{d}{=} \mathbf{R}. \tag{2.28}$$

From (2.19), (2.20), (2.22), (2.24), (2.26), (2.28) we obtain the following decomposition of the sojourn time of an arbitrary super customer:

$$\mathbf{R} \stackrel{d}{=} \mathbf{V}_{M/G/1} + \mathbf{B} + \mathbf{V}(\tilde{\mathbf{R}}), \tag{2.29}$$

all three terms in the right-hand side being independent. Note that the waiting time $\mathbf{W}$ and the service time $\mathbf{B}$ of a super customer are however *not* independent here, due to the dependence between the interarrival and service time. Namely, the larger the service time, the larger the interarrival time, and the smaller the waiting time. Hence, although the sojourn time is composed of the waiting time and the service time, (2.29) does *not* imply that $\mathbf{W} \stackrel{d}{=} \mathbf{V}_{M/G/1} + \mathbf{V}(\tilde{\mathbf{R}})$. In [40] it is shown how the joint distribution of $(\mathbf{W}, \mathbf{B})$

may be determined in a direct manner.
The sojourn time $\tilde{\mathbf{R}}$ of D may be thought of as that of an arbitrary super customer whose sojourn time is known to be smaller than an exponentially distributed interarrival time $\mathbf{A}$,

$$\mathrm{E}(\mathrm{e}^{-\omega\tilde{\mathbf{R}}}) = \frac{r(\gamma+\omega)}{r(\gamma)}, \qquad \mathrm{Re}\,\omega \geq 0. \tag{2.30}$$

Note that $r(\gamma)$ equals the probability that an arbitrary super customer finds the server idle upon arrival. Because of the PASTA property the latter probability equals the fraction of time that the server is idle, $1-\lambda\beta$.
So we obtain from (2.21), (2.23), (2.25), (2.29), and (2.30) the following functional equation for $r(\cdot)$:

$$r(\omega) = \frac{\gamma\omega}{\omega-\lambda(1-\beta(\omega))}\frac{r(\gamma+\lambda(1-\beta(\omega)))}{\gamma+\lambda(1-\beta(\omega))}, \qquad \mathrm{Re}\,\omega \geq 0. \tag{2.31}$$

We now solve the above functional equation.
Denote

$$f(\omega) := \frac{\gamma\omega}{\omega-\lambda(1-\beta(\omega))}, \qquad \mathrm{Re}\,\omega \geq 0,$$

$$g(\omega) := \gamma+\lambda(1-\beta(\omega)), \qquad \mathrm{Re}\,\omega \geq 0.$$

Then (2.31) may be rewritten as

$$r(\omega) = \frac{f(\omega)}{g(\omega)}r(g(\omega)), \qquad \mathrm{Re}\,\omega \geq 0. \tag{2.32}$$

Define

$$g^{(0)}(\omega) := \omega, \qquad \mathrm{Re}\,\omega \geq 0,$$

$$g^{(k)}(\omega) := g(g^{(k-1)}(\omega)), \qquad \mathrm{Re}\,\omega \geq 0, k = 1,2,\ldots.$$

Iterating (2.32) K times we find

$$r(\omega) = r(g^{(K+1)}(\omega))\prod_{k=0}^{K}\frac{f(g^{(k)}(\omega))}{g^{(k+1)}(\omega)}, \qquad \mathrm{Re}\,\omega \geq 0. \tag{2.33}$$

Letting $K \to \infty$ in (2.33),

$$r(\omega) = r(\omega^*)\prod_{k=0}^{\infty}\frac{f(g^{(k)}(\omega))}{g^{(k+1)}(\omega)}, \qquad \mathrm{Re}\,\omega \geq 0, \tag{2.34}$$

with $\omega^* := \lim_{K\to\infty} g^{(K)}(\omega)$. For a proof of the convergence for $\rho < 1$ we refer to [33], [40].

Putting $\omega = 0$ in (2.34),

$$r(\omega^*) = 1 \Big/ \prod_{k=0}^{\infty} \frac{f(g^{(k)}(0))}{g^{(k+1)}(0)} . \tag{2.35}$$

Substituting (2.35) back into (2.34),

$$r(\omega) = \prod_{k=0}^{\infty} \left[\frac{f(g^{(k)}(\omega))}{g^{(k+1)}(\omega)} \Big/ \frac{f(g^{(k)}(0))}{g^{(k+1)}(0)} \right], \qquad \text{Re}\,\omega \geq 0. \tag{2.36}$$

Above we showed how the work decomposition property could be exploited to derive a functional equation for the LST of the sojourn time distribution of an arbitrary super customer. We now show how similarly the queue length decomposition property can be applied to find an expression for the pgf of the joint distribution of the numbers of customers present at an arbitrary epoch. Similar to (2.17),

$$(\mathbf{N}_1, \ldots, \mathbf{N}_n) \stackrel{d}{=} (\mathbf{N}_1^*, \ldots, \mathbf{N}_n^*) + (\mathbf{N}_1^\#, \ldots, \mathbf{N}_n^\#), \tag{2.37}$$

with
$\mathbf{N}_i :=$ the number of type-i customers present at an arbitrary epoch;
$\mathbf{N}_i^* :=$ the number of released type-i customers present at an arbitrary epoch;
$\mathbf{N}_i^\# :=$ the number of remaining type-i customers being retained at an arbitrary epoch;
$(\mathbf{N}_1^*, \ldots, \mathbf{N}_n^*)$ and $(\mathbf{N}_1^\#, \ldots, \mathbf{N}_n^\#)$ being independent.
The joint distribution of $(\mathbf{N}_1^\#, \ldots, \mathbf{N}_n^\#)$ is given by

$$\mathrm{E}(z_1^{\mathbf{N}_1^\#} \ldots z_n^{\mathbf{N}_n^\#}) = \exp[-\sum_{i=1}^{n} \lambda_i (1 - z_i)(\tau_i + \sum_{j=i}^{n} \sigma_j)], \tag{2.38}$$

for $| z_i | \leq 1$, $i = 1, \ldots, n$.
Denote by $\mathbf{N}^*$ the total number of released customers present at an arbitrary epoch. Denote by $\mathbf{N}^0$ the total number of customers present at an arbitrary epoch in the corresponding system with a zero-delay collection procedure (i.e. with $\sigma_i = 0$, $\tau_i = 0$, $i = 1, \ldots, n$). Similar to (2.19),

$$\mathbf{N}^* \stackrel{d}{=} \mathbf{N}^0. \tag{2.39}$$

As described in Section 2.1, the following queue length decomposition property holds for $\mathbf{N}^0$, cf. [103]:

$$\mathbf{N}^0 \stackrel{d}{=} \mathbf{N}_{M/G/1} + \mathbf{N}_I^0, \tag{2.40}$$

with
$\mathbf{N}_{M/G/1} :=$ the total number of customers present at an arbitrary epoch in the corresponding $M/G/1$ system without a customer collection mechanism (i.e. where the customers have immediate access to the server);

$\mathbf{N}_I^0$:= the total number of customers present in the zero-delay system at an arbitrary epoch in a non-serving interval;
$\mathbf{N}_{M/G/1}$ and $\mathbf{N}_I^0$ being independent.
The distribution of $\mathbf{N}_{M/G/1}$ in (2.40) follows from the Pollaczek-Khintchine formula, cf. [73] p. 238,

$$\mathrm{E}(z^{\mathbf{N}_{M/G/1}}) = \frac{(1-\lambda\beta)(1-z)\beta(\lambda(1-z))}{\beta(\lambda(1-z))-z}, \qquad |\, z \,| \leq 1. \tag{2.41}$$

For any non-negative stochastic variable $\mathbf{T}$, denote by $\mathbf{N}(\mathbf{T})$ the total number of customers arriving to the system during a period of length $\mathbf{T}$, i.e.,

$$\mathrm{E}(z^{\mathbf{N}(\mathbf{T})}) = \mathrm{E}(\mathrm{e}^{-\lambda(1-z)\mathbf{T}}), \qquad |\, z \,| \leq 1. \tag{2.42}$$

Substituting (2.40) into (2.39), after decomposing the quantity $\mathbf{N}_I^0$ similarly to the quantity $\mathbf{V}_I^0$ in (2.22), we obtain the following detailed form of the queue length decomposition property:

$$\mathbf{N}^* \stackrel{d}{=} \mathbf{N}_{M/G/1} + \mathbf{N}(\mathbf{A}) + \mathbf{N}(\tilde{\mathbf{R}}). \tag{2.43}$$

As the order of service is assumed not to discriminate between the various customer types,

$$\mathrm{E}(z_1^{\mathbf{N}_1^*} \ldots z_n^{\mathbf{N}_n^*}) = \mathrm{E}(\pi(z)^{\mathbf{N}^*}), \qquad |\, z_i \,| \leq 1, \;\; i = 1, \ldots, n, \tag{2.44}$$

with $\pi(z) := \sum_{i=1}^{n} \lambda_i z_i / \lambda$.
From (2.37), (2.38), (2.44), (2.43) we obtain

$$\mathrm{E}(z_1^{\mathbf{N}_1} \ldots z_n^{\mathbf{N}_n}) = \mathrm{E}(\pi(z)^{\mathbf{N}_{M/G/1}})\mathrm{E}(\pi(z)^{\mathbf{N}(\mathbf{A})})\mathrm{E}(\pi(z)^{\mathbf{N}(\tilde{\mathbf{R}})}) \tag{2.45}$$
$$\exp[-\sum_{i=1}^{n} \lambda_i(1-z_i)(\tau_i + \sum_{j=i}^{n} \sigma_j)],$$

for $|\, z_i \,| \leq 1$, $i = 1, \ldots, n$.
From (2.42),

$$\mathrm{E}(z^{\mathbf{N}(\mathbf{A})}) = \frac{\gamma}{\gamma + \lambda(1-z)}, \qquad |\, z \,| \leq 1. \tag{2.46}$$

From (2.30), (2.42),

$$\mathrm{E}(z^{\mathbf{N}(\tilde{\mathbf{R}})}) = \frac{r(\gamma + \lambda(1-z))}{r(\gamma)}, \qquad |\, z \,| \leq 1, \tag{2.47}$$

with $r(\cdot)$ as in (2.36).
Thus $\mathrm{E}(z_1^{\mathbf{N}_1} \ldots z_n^{\mathbf{N}_n})$ is completely specified through (2.41), (2.45)-(2.47) and (2.36).

Taking $z = (1, \ldots, 1, y, 1, \ldots, 1)$ in (2.45) with y as i-th argument,

$$\mathrm{E}(y^{\mathbf{N}_i}) = \mathrm{E}(\pi_i(y)^{\mathbf{N}_{M/G/1}})\mathrm{E}(\pi_i(y)^{\mathbf{N}(\mathbf{A})})\mathrm{E}(\pi_i(y)^{\mathbf{N}(\tilde{\mathbf{R}})}) \exp[-\lambda_i(1-y)(\tau_i + \sum_{j=i}^{n}\sigma_j)], \tag{2.48}$$

with $\pi_i(y) := 1 - \lambda_i(1-y)/\lambda$.
Taking $\lambda_i(1-y) = \omega$ in (2.48), we obtain the following decomposition of the sojourn time $\mathbf{R}_i$ of an arbitrary type-i customer:

$$\mathrm{E}(\mathrm{e}^{-\omega\mathbf{R}_i}) = \mathrm{E}(\mathrm{e}^{-\omega\mathbf{R}_{M/G/1}})\mathrm{E}(\mathrm{e}^{-\omega\mathbf{A}})\mathrm{E}(\mathrm{e}^{-\omega\tilde{\mathbf{R}}}) \exp[-\omega(\tau_i + \sum_{j=i}^{n}\sigma_j)]. \tag{2.49}$$

(Recall that within the various customer classes the order of service is assumed to be FCFS.)
Noting that $\mathrm{E}(\mathrm{e}^{-\omega\mathbf{R}_i}) = \mathrm{E}(\mathrm{e}^{-\omega\mathbf{W}_i})\beta_i(\omega)$, we obtain from (2.49) the following decomposition of the waiting time $\mathbf{W}_i$ of an arbitrary type-i customer:

$$\mathrm{E}(\mathrm{e}^{-\omega\mathbf{W}_i}) = \frac{\beta(\omega)}{\beta_i(\omega)}\mathrm{E}(\mathrm{e}^{-\omega\mathbf{W}_{M/G/1}})\mathrm{E}(\mathrm{e}^{-\omega\mathbf{A}})\mathrm{E}(\mathrm{e}^{-\omega\tilde{\mathbf{R}}}) \exp[-\omega(\tau_i + \sum_{j=i}^{n}\sigma_j)]. \tag{2.50}$$

Chapter 3

Polling systems with zero and non-zero switch-over times

3.1 Introduction

In the present chapter we consider two different single-server polling systems: (i) a model with *zero* switch-over times, and (ii) a model with *non-zero* switch-over times, in which the server keeps cycling when the system is empty. For both models we relate the steady-state queue length distribution at a queue to the queue length distribution at visit beginning and visit completion instants at that queue. As a by-product we obtain a shorter proof of the Fuhrmann-Cooper decomposition, discussed before in Section 2.2. For the important class of service disciplines with a branching structure satisfying Property 1.4.1 defined in Section 1.4, we expose a strong relationship between both the queue length and the waiting-time distribution in the two models. We also show how the latter relationship can be exploited to reduce the computational complexity of numerical moment calculations.

As described in Section 2.1, polling systems may be viewed as queueing systems with service interruptions. Focusing on a specific queue in isolation, the service interruptions correspond to the intervisit times of the server with regard to that queue. Accordingly, the concept of queue length decomposition for queues with service interruptions (cf. Equation (2.1), Fuhrmann & Cooper [103]) has proven to be very fruitful for the analysis of polling models. It has also led to the concept of work decomposition in polling models (cf. Equation (2.10), Boxma [38]), which relates the amount of work in a system *with* switch-over times to the amount of work in a system with similar traffic characteristics but *without* switch-over times. Note that in the latter case the switch-over times constitute the service interruptions. Heretofore, models with switch-over times and models without switch-over times had usually been treated separately, of-

ten via different approaches; the problem with simply letting the switch-over times tend to zero in a polling model with non-zero switch-over times is that the number of polling epochs in an idle period tends to infinity, leading to degenerate distributions at such epochs, cf. [140], [86]. The relationship between the two models has further been exposed in some recent papers of Cooper, Niu, & Srinivasan [79], Fuhrmann [101], and Srinivasan, Niu, & Cooper [169]; these authors consider waiting times and queue lengths instead of workloads. In the present chapter we unify and generalize some of their results.

Firstly, in Section 3.3, we use a beautiful relation of Eisenberg [84] (see also [86]), which has received too little attention in the literature, to relate the probability generating functions (pgf's) of queue lengths at various instants in the polling system (visit beginnings and endings, service beginnings and endings). We observe that this relation, which was presented by Eisenberg [84] for the case of non-zero switch-over times, also holds for the case of zero switch-over times, and we show how it almost instantaneously gives a simple proof of the above-mentioned Fuhrmann-Cooper decomposition for the queue lengths at the various queues of a polling system.

Eisenberg's relation leads to an expression for the joint queue length pgf at service completion instants at some queue into the joint queue length pgf's at the beginning and the end of a visit to that queue. The latter pgf's can be easily related, and determined, for the important class of polling models in which the service discipline at each queue satisfies Property 1.4.1, which we restate here for the sake of completeness:

Property 3.1.1
If there are k_i customers present at Q_i at the start of a visit, then during the course of the visit, each of these k_i customers will effectively be replaced in an i.i.d. manner by a random population having pgf $h_i(z_1, \ldots, z_n)$, which may be any n-dimensional pgf.

Resing [159] (see also Fuhrmann [99]) has studied polling systems that satisfy this property; this includes the case of exhaustive or gated service at all queues, but it excludes the case of 1-limited service at any queue. As described in Section 1.4, for this class of polling systems, the joint queue length process at visit instants of a fixed queue is a so-called *multi-type branching process* with immigration. The theory of multi-type branching processes (cf. Athreya & Ney [13], Resing [158]) thus leads to an expression for the pgf of the joint queue length process at visit beginning (polling) instants. In Section 3.4, for models that satisfy Property 3.1.1, we use a slightly adapted version of the results of Resing [159] to relate the joint queue length pgf's at visit beginning and visit ending instants, and then to obtain those pgf's. The results expose a close similarity between the cases with and without switch-over times. In Section 3.5 we determine the steady-state marginal queue length pgf at Q_i, both for the model with and the model without switch-over times, and we relate the transforms for those two cases; similarly for the waiting-time Laplace-Stieltjes Transform (LST) at Q_i. In Section 3.6 we describe how the relationship between the

waiting times in the two models can be exploited to reduce the computational complexity of numerical moment calculations.

3.2 Model description

The model under consideration consists of n queues, $Q_1, \ldots, Q_n$, each of infinite capacity, attended by a single server S. For the specification of the arrival, service, and switch-over processes we refer to the description of the 'basic model' in Section 1.3.
The server visits the queues in strictly cyclic order, $Q_1, \ldots, Q_n$. We consider two versions of the model. In the first variant all the switch-over times are zero, i.e., $\sigma(\infty) = 1$, $s = 0$, $s^{(2)} = 0$. In the other variant at least one of the switch-over times is non-zero with some positive probability, i.e., $\sigma(\infty) < 1$, $s > 0$, $s^{(2)} > 0$.
In the model with *non-zero* switch-over times, the server keeps switching when the system becomes empty. In the model with *zero* switch-over times, when the system has become empty, the server makes a full cycle, i.e., passes all the queues once, and subsequently stops right before Q_1. All this requires zero time. When the first new customer arrives, the server cycles along the queues to that customer. The choice of Q_1 is arbitrary, but for the application of the theory of multi-type branching processes in Section 3.4 it will be necessary to fix one position. There we shall discuss this issue in more detail.
We assume the service disciplines to be non-idling, i.e, the server is not allowed to idle when there are customers present. For now we do not specify the service disciplines any further.

3.3 The joint queue length distribution at various epochs

Eisenberg [84] studies the model under consideration for the case of non-zero switch-over times and the exhaustive service discipline at all queues (while briefly discussing the case of gated service at all queues). He considers the following four quantities, with $\mathbf{N}$ denoting a vector of numbers of customers at $Q_1, \ldots, Q_n$ and N a realization:

$L_i(t, N)$:= number of service beginnings at Q_i in $(0, t)$ for which $\mathbf{N} = N$;
$M_i(t, N)$:= number of service completions at Q_i in $(0, t)$ for which $\mathbf{N} = N$;
$F_i(t, N)$:= number of visit beginnings at Q_i in $(0, t)$ for which $\mathbf{N} = N$;
$G_i(t, N)$:= number of visit completions at Q_i in $(0, t)$ for which $\mathbf{N} = N$;

In the case of a service or visit completion the state is defined as what exists immediately after the departure of the customer.
Eisenberg [84] now makes the crucial observation that each time a visit beginning or a service completion occurs, this coincides with either a service beginning or a visit completion. Hence

$$F_i(t,N) + M_i(t,N) = L_i(t,N) + G_i(t,N). \tag{3.1}$$

We observe that (3.1) not only holds for the case of non-zero switch-over times and exhaustive or gated service, but for any service discipline, and also for the case of zero switch-over times. Define the following equilibrium state probabilities for this polling model:

$L_i(N) := \Pr(\mathbf{N} = N,\ S \text{ is at } Q_i \mid \text{service beginning instant})$;
$M_i(N) := \Pr(\mathbf{N} = N,\ S \text{ is at } Q_i \mid \text{service completion instant})$;
$F_i(N) := \Pr(\mathbf{N} = N \mid \text{visit beginning at } Q_i)$;
$G_i(N) := \Pr(\mathbf{N} = N \mid \text{visit completion at } Q_i)$.

Eisenberg [84] divides all four terms in (3.1) by the total number of service completions at all queues in $(0,t)$, and takes the limit for $t \to \infty$. He thus relates those four equilibrium state probabilities:

$$\gamma_i F_i(N) + M_i(N) = L_i(N) + \gamma_i G_i(N).$$

Here γ_i is the long-term ratio of the number of visit completions at Q_i to the number of customers that are handled by the system; in this cyclic polling model $\gamma_i \equiv \gamma$, $i = 1, \ldots, n$. Written in terms of pgf's,

$$\gamma F_i(z) + M_i(z) = L_i(z) + \gamma G_i(z), \tag{3.2}$$

for $z = (z_1, \ldots, z_n)$, $\mid z_j \mid \leq 1$, $j = 1, \ldots, n$; here $F_i(z)$ and $G_i(z)$ denote the pgf of the joint queue length distribution at visit beginnings and visit completions of Q_i, while $L_i(z)$ and $M_i(z)$ denote the pgf of the joint distribution of queue length vector and server position at service beginnings and service completions. Now Eisenberg observes that $M_i(z)$ and $L_i(z)$ are related via

$$M_i(z) = L_i(z)\beta_i(\sum_{j=1}^{n} \lambda_j(1 - z_j))/z_i, \tag{3.3}$$

for $\mid z_j \mid \leq 1$, $j = 1, \ldots, n$.
It follows from (3.2) and (3.3) that

$$M_i(z) = \frac{\gamma\beta_i(\sum_{j=1}^{n} \lambda_j(1 - z_j))}{z_i - \beta_i(\sum_{j=1}^{n} \lambda_j(1 - z_j))}[F_i(z) - G_i(z)]. \tag{3.4}$$

Eisenberg, considering the variant with switch-over times and exhaustive service, subsequently expresses $F_i(z)$ into $G_{i-1}(z)$. For the moment we refrain from that (see Section 3.4), but we observe that formula (3.4) is generally valid for the class of polling systems described in Section 3.2 (with and without switch-over times).
Taking $z = (1, \ldots, 1, y, 1, \ldots, 1)$ in (3.4), with y as i-th argument, and dividing by the probability λ_i/λ that an arbitrary service completion is at Q_i, gives

the queue length pgf at Q_i at a service completion instant at Q_i. A standard up- and down-crossing argument combined with PASTA shows that the queue length distribution at Q_i at its service completion instants, at its customer arrival instants and in steady-state are all the same. Hence, with $\mathbf{N}_i$ the steady-state queue length at Q_i and with $\mathbf{X}_i$ and $\mathbf{Y}_i$ the steady-state queue lengths at Q_i at the beginning and end of a visit at that queue: for $| y | \leq 1$,

$$\mathrm{E}(y^{\mathbf{N}_i}) = \frac{\lambda}{\lambda_i} \frac{\gamma\beta_i(\lambda_i(1-y))}{y - \beta_i(\lambda_i(1-y))}[\mathrm{E}(y^{\mathbf{X}_i}) - \mathrm{E}(y^{\mathbf{Y}_i})]. \tag{3.5}$$

Note that $1/\gamma$ equals the mean number of customers served per cycle, hence also the mean number of customers that arrive per cycle: $1/\gamma = \lambda \mathbf{EC}$, with $\mathbf{EC}$ the mean length of one cycle of S along the queues (a cycle w.r.t. Q_i is defined as the period between the start of two successive visits to Q_i; it is easily seen that the mean cycle time is the same for all i). Because S spends on the average a fraction ρ_i of a cycle at Q_i, we can write:

$$\mathrm{E}\mathbf{X}_i - \mathrm{E}\mathbf{Y}_i = \lambda_i(1-\rho_i)\mathbf{EC} = \frac{\lambda_i(1-\rho_i)}{\lambda\gamma}. \tag{3.6}$$

From (3.5) and (3.6), for $| y | \leq 1$:

$$\mathrm{E}(y^{\mathbf{N}_i}) = \frac{(1-\rho_i)(1-y)\beta_i(\lambda_i(1-y))}{\beta_i(\lambda_i(1-y)) - y} \frac{\mathrm{E}(y^{\mathbf{Y}_i}) - \mathrm{E}(y^{\mathbf{X}_i})}{(1-y)(\mathrm{E}\mathbf{X}_i - \mathrm{E}\mathbf{Y}_i)}. \tag{3.7}$$

The first term in the right-hand side denotes the pgf $\mathrm{E}(y^{\mathbf{N}_{i|M/G/1}})$ of the queue length distribution in the 'corresponding' isolated $M/G/1$ queue of Q_i with arrival rate λ_i and service time distribution LST $\beta_i(\cdot)$. Now consider the second term. Observe that $\mathbf{Y}_i$ not only denotes the queue length at Q_i at the end of a visit to that queue, but also the queue length at Q_i at the beginning of an intervisit period for that queue; while $\mathbf{X}_i$ denotes the queue length at Q_i at the *end* of such an intervisit period. Introducing $\mathbf{N}_{i|I}$, a stochastic variable with distribution the queue length distribution at an arbitrary instant in an intervisit period of Q_i, we have from Lemma 2.2.1:

$$\mathrm{E}(y^{\mathbf{N}_{i|I}}) = \frac{\mathrm{E}(y^{\mathbf{Y}_i}) - \mathrm{E}(y^{\mathbf{X}_i})}{(1-y)(\mathrm{E}\mathbf{X}_i - \mathrm{E}\mathbf{Y}_i)}. \tag{3.8}$$

This relation appears in the polling literature for various special cases (e.g., for exhaustive vacation models, where $\mathbf{Y}_i \equiv 0$). It holds also for non-Poisson arrivals, when $\mathbf{N}_{i|I}$ is defined as the queue length that is observed by an arbitrary customer that arrives at Q_i during an intervisit period.
Formulae (3.7) and (3.8) together yield the well-known Fuhrmann-Cooper queue length decomposition [103], applied to a queue in a polling model with or without switch-over times: for $| y | \leq 1$,

$$\mathrm{E}(y^{\mathbf{N}_i}) = \mathrm{E}(y^{\mathbf{N}_{i|M/G/1}})\mathrm{E}(y^{\mathbf{N}_{i|I}}). \tag{3.9}$$

Remark 3.3.1
Fuhrmann and Cooper [103] state five conditions under which their decomposition holds:
(i). Customers arrive at Q_i according to a Poisson process of rate λ_i.
(ii). All customers arriving at Q_i are eventually served.
(iii). Customers enter service in an order that is independent of their service times.
(iv). Service is non-preemptive.
(v). The rules that govern when the server begins and ends visit periods to Q_i do not anticipate future jumps of the Poisson arrival process at Q_i.
These assumptions indeed hold in our polling model, and are implicitly used in the derivation of (3.9). The above proof, with as key steps (3.1), (3.3) and (3.8), in fact also holds for vacation models without a polling context. Note that the relations $1/\gamma = \lambda \mathbf{EC}$ and (3.6) hold generally for queues with some vacation (intervisit) mechanism. We refer to Keilson & Servi [125] for another short proof of the Fuhrmann-Cooper decomposition.

□

The waiting-time LST at Q_i immediately follows from (3.9), when we assume that within each of the queues customers are served in order of arrival. Denote by $\mathbf{W}_i$ the waiting time of an arbitrary type-i customer. Denote by $\mathbf{W}_{i|M/G/1}$ the waiting time of an arbitrary customer in the 'corresponding' isolated $M/G/1$ queue of Q_i. By the distributional form of Little's law, cf. Keilson & Servi [125], similar to equation (2.9) in Section 2.2,

$$\mathrm{E}(\mathrm{e}^{-\omega \mathbf{W}_i}) = \mathrm{E}(\mathrm{e}^{-\omega \mathbf{W}_{i|M/G/1}}) \mathrm{E}((1-\omega/\lambda_i)^{\mathbf{N}_{i|I}}). \tag{3.10}$$

In Section 3.5 we shall return to this relation, for the case of polling models that satisfy Property 3.1.1.

3.4 The joint queue length distribution at polling epochs

In the previous section we have seen that Eisenberg's results [84] yield simple relations between the pgf $M_i(z)$ of the joint queue length vector at service completion epochs (or $L_i(z)$, at service beginning epochs) and the pgf's $F_i(z)$ and $G_i(z)$ of the joint queue length vector at visit beginning and visit completion epochs. We now restrict ourselves to polling models for which the service discipline at each queue satisfies Property 3.1.1. Property 3.1.1 prescribes how each of the customers present at Q_i at the visit beginning is replaced by independent families of customers at its visit completion. This enables one to express $G_i(\cdot)$ nicely into $F_i(\cdot)$, and to finally determine each of the functions $F_i(\cdot)$ (after which the pgf's $G_i(\cdot)$, $M_i(\cdot)$ and $L_i(\cdot)$ follow). In our analysis we follow Resing [159].
First some words on the ergodicity conditions. In the sequel we assume that

$\rho < 1$ and $s_i < \infty$ for all i. Resing [159] proves that for the subclass of so-called Bernoulli-type service disciplines, including exhaustive and gated service, cf. Section 1.3, these conditions together constitute sufficient ergodicity conditions. His proof is based on the observation that for these Bernoulli-type service disciplines the derivatives of $h_i(z_1, \ldots, z_n)$ take the form $\frac{\partial h_i(z)}{\partial z_j} \mid_{z=1} = \lambda_j \alpha_i \frac{\beta_i}{1-\rho_i}$, $i \neq j$, $\frac{\partial h_i(z)}{\partial z_i} \mid_{z=1} = 1 - \alpha_i$, with α_i some coefficient in $(0, 1]$ determined by the parameters of Q_i. It may be easily verified however that the latter form of the derivatives applies for *any* non-idling service discipline that satisfies Property 3.1.1 with $h_i(z_1, \ldots, z_n) \neq z_i$. As the proof in Resing [159] further does not rely on the specific form of α_i, we may conclude that $\rho < 1$ and $s_i < \infty$ for all i together constitute sufficient ergodicity conditions for *any* non-idling service discipline that satisfies Property 3.1.1 with $h_i(z_1, \ldots, z_n) \neq z_i$.
Property 3.1.1 implies that

$$G_i(z) = F_i(z_1, \ldots, z_{i-1}, h_i(z), z_{i+1}, \ldots, z_n). \tag{3.11}$$

In the case of gated service $h_i(z)$ is simply the pgf of the joint distribution of the numbers of arrivals at all queues during one service time at Q_i: $h_i(z) = \beta_i(\sum_{j=1}^{n} \lambda_j(1 - z_j))$.
In the case of exhaustive service: $h_i(z) = \eta_i(\sum_{j \neq i} \lambda_j(1 - z_j))$, with $\eta_i(\cdot)$ the LST of the length of the busy period in the 'corresponding' isolated $M/G/1$ queue of Q_i.
Next we relate $F_i(z)$ to $G_{i-1}(z)$.
In the case of non-zero switch-over times:

$$F_i(z) = G_{i-1}(z)\sigma_{i-1}(\sum_{j=1}^{n} \lambda_j(1 - z_j)). \tag{3.12}$$

In the case of zero switch-over times (in the sequel we add a superscript 0 for that case, to distinguish its quantities from those for non-zero switch-over times):

$$F_i^0(z) = G_{i-1}^0(z), \tag{3.13}$$

for $i = 2, \ldots, n$. The relation between $F_1^0(z)$ and $G_n^0(z)$ deserves special attention, because of our convention, mentioned in Section 3.2, concerning the behavior of the server in an empty system. When all queues in the model with zero switch-over times have become empty, in our convention S makes a full cycle and subsequently stops right before Q_1 (all this requires zero time). When the first new customer arrives, S cycles along the queues to that customer. The consequence of this is that when the system is empty at the start of a visit to Q_1, then the next visit to Q_1 does not take place until a customer has arrived. We can write

$$F_1^0(z) = G_n^0(z) - F_1^0(0)[1 - g^0(z)], \tag{3.14}$$

with

$$g^0(z) := \sum_{i=1}^{n} \frac{\lambda_i}{\lambda} z_i.$$

Here $g^0(\cdot)$ represents the 'immigration process' of the multi-type branching process: it is the pgf of the arrival process of customers during periods in which the system is empty.
Substituting (3.11) into (3.12), respectively (3.13) and (3.14), we can relate $F_i(\cdot)$ to $F_{i-1}(\cdot)$. We distinguish between the two cases of zero and non-zero switch-over times. In both cases the following branching functions play a crucial role, thus establishing the link between both cases.
Define

$$f(z) := (f_1(z), \ldots, f_n(z)), \tag{3.15}$$

with

$$f_i(z) := h_i(z_1, \ldots, z_i, f_{i+1}(z), \ldots, f_n(z)) \tag{3.16}$$

for $\mid z_j \mid \leq 1$, $j = 1, \ldots, n$. This is the *offspring* pgf, the pgf of the joint distribution of the numbers of customers at the end of a cycle w.r.t. Q_1 that are *descendants* of a type-i customer. In this branching process setting, a descendant of some customer K is a customer that has arrived during the service time of K or of one of its descendants.
Define

$$f^{(0)}(z) := z,$$

$$f^{(k)}(z) := f(f^{(k-1)}(z)), \qquad k \geq 1,$$

for $\mid z_j \mid \leq 1$, $j = 1, \ldots, n$.

Case I: Zero switch-over times
Substituting (3.11) into (3.13),

$$F_i^0(z) = F_{i-1}^0(z_1, \ldots, z_{i-2}, h_{i-1}(z), z_i, \ldots, z_n) \tag{3.17}$$

for $i = 2, \ldots, n$. Starting from (3.14) and (3.11) for $i = n$, and subsequently using (3.17) for $i = n, n-1, \ldots, 2$, one obtains

$$F_1^0(z) = F_1^0(f(z)) - F_1^0(0)[1 - g^0(z)]. \tag{3.18}$$

Iterating (3.18) yields

$$F_1^0(z) = 1 - F_1^0(0) \sum_{k=1}^{\infty} [1 - g^0(f^{(k)}(z))], \tag{3.19}$$

with

$$F_1^0(0) = \left[1 + \sum_{k=1}^{\infty}[1 - g^0(f^{(k)}(0))]\right]^{-1},$$

the infinite sum being convergent when the ergodicity conditions are fulfilled.

Introduce, for $| z_j | \leq 1$, $j = 1, \ldots, n$,

$$H(z) := \sum_{k=1}^{\infty}\sum_{i=1}^{n} \lambda_i (1 - f_i^{(k)}(z)). \tag{3.20}$$

Then we can write

$$\begin{aligned} F_1^0(z) &= 1 - F_1^0(0) \sum_{k=1}^{\infty}\sum_{i=1}^{n} \frac{\lambda_i}{\lambda}(1 - f_i^{(k)}(z)) \\ &= 1 - F_1^0(0) H(z)/\lambda, \end{aligned} \tag{3.21}$$

with

$$F_1^0(0) = [1 + H(0)/\lambda]^{-1}.$$

Remark 3.4.1
Although we shall sometimes find it convenient to concentrate on Q_1, it should be noted that our convention for the position of S in an empty system does not affect the waiting-time and queue length distributions.
In fact our convention slightly differs from that of Resing [159], who assumes that in an empty system S immediately stops right *behind* Q_1 and hence takes $g^0(z) = \sum_{i=1}^{n} \frac{\lambda_i}{\lambda} f_i(z)$. Our convention enables us to simultaneously apply the theory of multi-type branching processes and Eisenberg's approach.

□

Case II: Non-zero switch-over times
Substituting (3.11) into (3.12),

$$F_i(z) = F_{i-1}(z_1, \ldots, z_{i-2}, h_{i-1}(z), z_i, \ldots, z_n)\sigma_{i-1}(\sum_{j=1}^{n} \lambda_j (1 - z_j)). \tag{3.22}$$

Applying (3.22) n times (which corresponds to following the server during one full cycle w.r.t. Q_1),

$$F_1(z) = F_1(f(z))g(z), \tag{3.23}$$

with

$$g(z) = \prod_{i=1}^{n} \sigma_i(\sum_{j=1}^{i} \lambda_j (1 - z_j) + \sum_{j=i+1}^{n} \lambda_j (1 - f_j(z))).$$

Here $g(\cdot)$ represents the 'immigration process' of this multi-type branching process: it is the pgf of the vector of all customers that either have arrived in the switch-over periods of the past cycle (measured w.r.t. Q_1), or are descendants of such customers.
Iterating (3.23) yields

$$\begin{aligned} F_1(z) &= \prod_{k=1}^{\infty} g(f^{(k)}(z)) \qquad (3.24) \\ &= \prod_{k=1}^{\infty}\prod_{i=1}^{n} \sigma_i(\sum_{j=1}^{i} \lambda_j(1-f_j^{(k)}(z)) + \sum_{j=i+1}^{n} \lambda_j(1-f_j^{(k+1)}(z))), \end{aligned}$$

the infinite product being convergent when the ergodicity conditions are fulfilled.

It is clear from (3.21) and (3.24) that $F_1(z)$ as well as $F_1^0(z)$ is determined by $\sum_{j=1}^{n} \lambda_j(1-f_j^{(k)}(z))$. For *constant switch-over times* the connection becomes even closer, as (3.24) reduces to a simpler expression, which we present for future reference:

$$F_1(z) = \qquad (3.25)$$

$$\exp[-\sum_{k=1}^{\infty}\sum_{i=1}^{n} s_i\{\sum_{j=1}^{i} \lambda_j(1-f_j^{(k)}(z)) + \sum_{j=i+1}^{n} \lambda_j(1-f_j^{(k+1)}(z))\}].$$

Using (3.20), we can rewrite (3.25) into

$$F_1(z) = \exp[-sH(z) + \sum_{j=1}^{n} \lambda_j(1-z_j)\sum_{i=1}^{j-1} s_i]. \qquad (3.26)$$

3.5 Marginal queue lengths and waiting times

In the previous section the queue length pgf's $F_i(z)$ and $F_i^0(z)$ at visit beginning instants have been determined for the class of cyclic polling models in which Property 3.1.1 holds for all service disciplines. In Section 3.3 we already obtained a decomposition for the pgf of the marginal queue length distribution at Q_i, and for the waiting-time LST at Q_i, into a corresponding $M/G/1$ term and a term involving $\mathrm{E}(y^{\mathbf{X}_i})$ and $\mathrm{E}(y^{\mathbf{Y}_i})$ (via the pgf $\mathrm{E}(y^{\mathbf{N}_{i|I}})$). In particular, denoting

$\tilde{h}_i(y) := h_i(1,\ldots,1,y,1,\ldots,1);$
$\tilde{F}_i(y) := F_i(1,\ldots,1,y,1,\ldots,1);$
$\tilde{F}_i^0(y) := F_i^0(1,\ldots,1,y,1,\ldots,1),$

with y as i-th argument, it follows from (3.8) and (3.11) for the case of non-zero switch-over times that

$$\mathrm{E}(y^{\mathbf{N}_{i|I}}) = \frac{\tilde{F}_i(\tilde{h}_i(y)) - \tilde{F}_i(y)}{(1-y)\tilde{F}_i'(1)(1-\tilde{h}_i'(1))}; \tag{3.27}$$

the same result holds for the case of zero switch-over times, replacing $\tilde{F}_i(\cdot)$ by $\tilde{F}_i^0(\cdot)$ in (3.27). Similarly indicating queue lengths, and waiting times, by a superscript 0 in the case of zero times, we find from (3.9) and (3.10):

$$\mathrm{E}(y^{\mathbf{N}_i}) = \mathrm{E}(y^{\mathbf{N}_i^0}) \frac{[\tilde{F}_i(\tilde{h}_i(y)) - \tilde{F}_i(y)]\tilde{F}_i^{0\prime}(1)}{[\tilde{F}_i^0(\tilde{h}_i(y)) - \tilde{F}_i^0(y)]\tilde{F}_i'(1)}, \tag{3.28}$$

$$\mathrm{E}(\mathrm{e}^{-\omega\mathbf{W}_i}) = \mathrm{E}(\mathrm{e}^{-\omega\mathbf{W}_i^0}) \frac{[\tilde{F}_i(\tilde{h}_i(1-\omega/\lambda_i)) - \tilde{F}_i(1-\omega/\lambda_i)]\tilde{F}_i^{0\prime}(1)}{[\tilde{F}_i^0(\tilde{h}_i(1-\omega/\lambda_i)) - \tilde{F}_i^0(1-\omega/\lambda_i)]\tilde{F}_i'(1)}. \tag{3.29}$$

For exhaustive service $\tilde{h}_i(\cdot) \equiv 1$; for gated service $\tilde{h}_i(y) = \beta_i(\lambda_i(1-y))$. Differentiating (3.10) and (3.29) once, putting $\omega = 0$,

$$\mathrm{E}\mathbf{W}_i = \frac{\lambda_i\beta_i^{(2)}}{2(1-\lambda_i\beta_i)} + \frac{\tilde{F}_i''(1)}{2\tilde{F}_i'(1)\lambda_i}(1+\tilde{h}_i'(1)) - \frac{\tilde{h}_i''(1)}{2(1-\tilde{h}_i'(1))\lambda_i}, \tag{3.30}$$

and

$$\mathrm{E}\mathbf{W}_i - \mathrm{E}\mathbf{W}_i^0 = [\frac{\tilde{F}_i''(1)}{2\tilde{F}_i'(1)\lambda_i} - \frac{\tilde{F}_i^{0\prime\prime}(1)}{2\tilde{F}_i^{0\prime}(1)\lambda_i}](1+\tilde{h}_i'(1)). \tag{3.31}$$

Let us now (without loss of generality) concentrate on $\mathbf{W}_1$ and $\mathbf{W}_1^0$. Denoting

$\tilde{f}_i^{(k)}(y) := f_i^{(k)}(y, 1, \ldots, 1)$,

it follows from (3.21) that

$$\tilde{F}_1^0(y) = 1 - F_1^0(0)\tilde{H}(y)/\lambda, \tag{3.32}$$

with

$$\tilde{H}(y) := \sum_{k=1}^{\infty}\sum_{i=1}^{n} \lambda_i(1 - \tilde{f}_i^{(k)}(y)). \tag{3.33}$$

In particular, $\tilde{F}_1^{0\prime}(1) = -F_1^0(0)\tilde{H}'(1)/\lambda$. It follows from (3.26) and (3.33) that $\tilde{F}_1'(1) = -s\tilde{H}'(1)$, and hence

$$\tilde{F}_1'(1) = \tilde{F}_1^{0\prime}(1)s\lambda/F_1^0(0). \tag{3.34}$$

For exhaustive service, from (3.29), (3.32), (3.34),

Corollary 3.5.1

$$\mathrm{E}(\mathrm{e}^{-\omega\mathbf{W}_1}) = \mathrm{E}(\mathrm{e}^{-\omega\mathbf{W}_1^0}) \frac{1-\tilde{F}_1(1-\omega/\lambda_1)}{s\tilde{H}(1-\omega/\lambda_1)}, \tag{3.35}$$

which is Theorem 1 in Srinivasan et al. [169].

For constant switch-over times, it follows from (3.21) and (3.26) that

$$\tilde{F}_1(y) = \exp[-s\tilde{H}(y)], \tag{3.36}$$

and hence we have

Corollary 3.5.2

$$\mathrm{E}(\mathrm{e}^{-\omega \mathbf{W}_1}) = \tag{3.37}$$

$$\mathrm{E}(\mathrm{e}^{-\omega \mathbf{W}_1^0}) \frac{\exp[-s\tilde{H}(\tilde{h}_1(1-\omega/\lambda_1))] - \exp[-s\tilde{H}(1-\omega/\lambda_1)]}{s[\tilde{H}(1-\omega/\lambda_1) - \tilde{H}(\tilde{h}_1(1-\omega/\lambda_1))]},$$

which generalizes Theorem 2 in Srinivasan et al. [169] (there the result is obtained for exhaustive service, with $\tilde{h}_1(\cdot) \equiv 1$).

Remark 3.5.1 Substituting $\sigma_i(\omega) = 1 - s_i\omega + o(s_i)$ for $s_i \to 0$, $i = 1, \ldots, n$, in (3.24) yields that

$$\tilde{F}_1(y) =$$

$$1 - \sum_{i=1}^{n} s_i \sum_{k=1}^{\infty} \left[\sum_{j=1}^{i} \lambda_j (1 - \tilde{f}_j^{(k)}(y)) + \sum_{j=i+1}^{n} \lambda_j (1 - \tilde{f}_j^{(k+1)}(y)) \right] + o(s) =$$

$$1 - s\tilde{H}(y) + o(s).$$

Hence, for $s \to 0$ in (3.35), $\mathbf{W}_1$ approaches $\mathbf{W}_1^0$ in distribution. Using (3.21) and (3.29), the same statement follows for other service disciplines satisfying Property 3.1.1.

□

3.6 Computational aspects

In this last section we describe how the relationship between the waiting times in the models with and without switch-over times may be exploited to reduce the computational complexity of moment calculations. For ease of presentation we focus on the first moment, but similar observations hold for the higher moments.

To determine $\mathrm{E}\mathbf{W}_1$ and $\mathrm{E}\mathbf{W}_1^0$ according to formulae (3.30) and (3.31), we need to compute the quantities $\tilde{F}_1'(1)$, $\tilde{F}_1''(1)$, $\tilde{F}_1^{0\prime}(1)$, and $\tilde{F}_1^{0\prime\prime}(1)$. The quantities $\tilde{h}_1'(1)$ and $\tilde{h}_1''(1)$ occurring in (3.30) and (3.31) may simply be determined from the service discipline at Q_1.

We first introduce some notation. Denote $\phi_i^{(k)} := \frac{d}{dy}\tilde{f}_i^{(k)}(y)_{|y=1}$, $\psi_i^{(k)} :=$

$\frac{d^2}{dy^2}\tilde{f}_i^{(k)}(y)_{|y=1}$, $i = 1,\ldots,n$, $k = 1,\ldots,\infty$. Define $\Phi_i := \sum_{k=1}^{\infty} \phi_i^{(k)}$, $\Psi_i := \sum_{k=1}^{\infty} \psi_i^{(k)}$, $i = 1,\ldots,n$, $\Phi := \sum_{i=1}^{n} \lambda_i \Phi_i$, $\Psi := \sum_{i=1}^{n} \lambda_i \Psi_i$.
Differentiating (3.32),

$$\tilde{F}_1^{0\prime}(1) = \frac{F_1^0(0)}{\lambda} \sum_{k=1}^{\infty} \sum_{i=1}^{n} \lambda_i \phi_i^{(k)} = \frac{F_1^0(0)}{\lambda} \Phi; \tag{3.38}$$

$$\tilde{F}_1^{0\prime\prime}(1) = \frac{F_1^0(0)}{\lambda} \sum_{k=1}^{\infty} \sum_{i=1}^{n} \lambda_i \psi_i^{(k)} = \frac{F_1^0(0)}{\lambda} \Psi. \tag{3.39}$$

Taking $z = (y, 1, \ldots, 1)$ in (3.24), differentiating w.r.t. to y,

$$\tilde{F}_1'(1) = s \sum_{k=1}^{\infty} \sum_{i=1}^{n} \lambda_i \phi_i^{(k)} = s\Phi; \tag{3.40}$$

$$\begin{aligned} \tilde{F}_1''(1) &= \left(\tilde{F}_1'(1)\right)^2 + s \sum_{k=1}^{\infty} \sum_{i=1}^{n} \lambda_i \psi_i^{(k)} \\ &+ \sum_{k=1}^{\infty} \sum_{i=1}^{n} \left(s_i^{(2)} - s_i^2\right) \left(\sum_{j=1}^{i} \lambda_j \phi_j^{(k)} + \sum_{j=i+1}^{n} \lambda_j \phi_j^{(k+1)} \right)^2 \\ &= \left(\tilde{F}_1'(1)\right)^2 + s\Psi + \sum_{i=1}^{n} \left(s_i^{(2)} - s_i^2\right) \chi_i, \end{aligned} \tag{3.41}$$

with $\chi_i = \sum_{k=1}^{\infty} \left(\sum_{j=1}^{i} \lambda_j \phi_j^{(k)} + \sum_{j=i+1}^{n} \lambda_j \phi_j^{(k+1)} \right)^2$.
The quantities Φ, Ψ, and χ_i, $i = 1,\ldots,n$, may be computed as follows. Taking $z = (y, 1, \ldots, 1)$ in (3.16),

$$\tilde{f}_i^{(k+1)}(y) = h_i(\tilde{f}_1^{(k)}(y), \ldots, \tilde{f}_i^{(k)}(y), \tilde{f}_{i+1}^{(k+1)}(y), \ldots, \tilde{f}_n^{(k+1)}(y)). \tag{3.42}$$

Define $\mathbf{T}_i$ as the visit time at Q_i generated by an arbitrary type-i customer present at the start of a visit to Q_i. Then $\frac{\partial}{\partial z_j} h_i(z)_{|z=1} = \lambda_j \mathrm{E}\mathbf{T}_i$, $\frac{\partial^2}{\partial z_j \partial z_l} h_i(z)_{|z=1} = \lambda_j \lambda_l \mathrm{E}(\mathbf{T}_i^2)$, $j \neq i$, $l \neq i$. Define $\mathbf{V}_i$ as the total visit time at Q_i in a cycle. From $\mathrm{E}\mathbf{V}_i = \mathrm{E}\mathbf{X}_i \mathrm{E}\mathbf{T}_i$, $\mathrm{E}\mathbf{X}_i = \mathrm{E}\mathbf{Y}_i + \lambda_i(1-\rho_i)\mathrm{E}\mathbf{C}$, $\mathrm{E}\mathbf{V}_i = \rho_i \mathrm{E}\mathbf{C}$, $\mathrm{E}\mathbf{Y}_i = \mathrm{E}\mathbf{X}_i \frac{\partial}{\partial z_i} h_i(z)_{|z=1}$, it immediately follows that $\frac{\partial}{\partial z_i} h_i(z)_{|z=1} = (\lambda_i - 1/\beta_i)\mathrm{E}\mathbf{T}_i + 1$. Thus, differentiating (3.42),

$$\begin{aligned} \phi_i^{(k+1)} &= \sum_{j=1}^{i} \frac{\partial}{\partial z_j} h_i(z)_{|z=1} \phi_j^{(k)} + \sum_{j=i+1}^{n} \frac{\partial}{\partial z_j} h_i(z)_{|z=1} \phi_j^{(k+1)} \\ &= \mathrm{E}\mathbf{T}_i \left[\sum_{j=1}^{i} \lambda_j \phi_j^{(k)} + \phi_i^{(k)} / \mathrm{E}\mathbf{T}_i - \phi_i^{(k)} / \beta_i + \sum_{j=i+1}^{n} \lambda_j \phi_j^{(k+1)} \right]; \end{aligned} \tag{3.43}$$

$$\begin{aligned}
\psi_i^{(k+1)} &= \sum_{j=1}^{i} \frac{\partial}{\partial z_j} h_i(z)_{|z=1} \psi_j^{(k)} + \sum_{j=i+1}^{n} \frac{\partial}{\partial z_j} h_i(z)_{|z=1} \psi_j^{(k+1)} \qquad (3.44)\\
&+ \sum_{j=1}^{i} \sum_{l=1}^{i} \frac{\partial^2}{\partial z_j \partial z_l} h_i(z)_{|z=1} \phi_j^{(k)} \phi_l^{(k)}\\
&+ 2 \sum_{j=1}^{i} \sum_{l=i+1}^{n} \frac{\partial^2}{\partial z_j \partial z_l} h_i(z)_{|z=1} \phi_j^{(k)} \phi_l^{(k+1)}\\
&+ \sum_{j=i+1}^{n} \sum_{l=i+1}^{n} \frac{\partial^2}{\partial z_j \partial z_l} h_i(z)_{|z=1} \phi_j^{(k+1)} \phi_l^{(k+1)}\\
&= \mathrm{E}\mathbf{T}_i \left[\sum_{j=1}^{i} \lambda_j \psi_j^{(k)} + \psi_i^{(k)} / \mathrm{E}\mathbf{T}_i - \psi_i^{(k)} / \beta_i + \sum_{j=i+1}^{n} \lambda_j \psi_j^{(k+1)} \right]\\
&+ \mathrm{E}(\mathbf{T}_i^2) \left(\sum_{j=1}^{i} \lambda_j \phi_j^{(k)} + \sum_{j=i+1}^{n} \lambda_j \phi_j^{(k+1)} \right)^2\\
&+ \phi_i^{(k)} \left[2 \sum_{j=1}^{i-1} \tau_{ij} \phi_j^{(k)} + \tau_{ii} \phi_i^{(k)} + 2 \sum_{j=i+1}^{n} \tau_{ij} \phi_j^{(k+1)} \right],
\end{aligned}$$

with $\tau_{ij} := \frac{\partial^2}{\partial z_i z_j} h_i(z)_{|z=1} - \lambda_i \lambda_j \mathrm{E}(\mathbf{T}_i^2)$.
Summing (3.43) and (3.44) over $k = 1, \ldots, \infty$, we obtain

$$\frac{\Phi_i}{\beta_i} - \frac{\phi_i^{(0)}}{\mathrm{E}\mathbf{T}_i} + \sum_{j=i+1}^{n} \lambda_j \phi_j^{(0)} = \Phi, \qquad (3.45)$$

and

$$\frac{\Psi_i}{\beta_i} - \frac{\psi_i^{(0)} + \mathrm{E}(\mathbf{T}_i^2)\chi_i + \xi_i}{\mathrm{E}\mathbf{T}_i} + \sum_{j=i+1}^{n} \lambda_j \psi_j^{(0)} = \Psi, \qquad (3.46)$$

with $\xi_i = \sum_{k=1}^{\infty} \phi_i^{(k)} \left[2 \sum_{j=1}^{i-1} \tau_{ij} \phi_j^{(k)} + \tau_{ii} \phi_i^{(k)} + 2 \sum_{j=i+1}^{n} \tau_{ij} \phi_j^{(k+1)} \right]$.
Multiplying (3.45) and (3.46) by ρ_i, summing over $i = 1, \ldots, n$, using $\phi_1^{(0)} = 1$, $\phi_i^{(0)} = 0$, $i = 2, \ldots, n$, $\psi_i^{(0)} = 0$, $i = 1, \ldots, n$, we obtain

$$\Phi = \frac{\rho_1 / \mathrm{E}\mathbf{T}_1}{1 - \rho}, \qquad (3.47)$$

and

$$\Psi = \frac{\sum_{i=1}^{n} \rho_i \left[\mathrm{E}(\mathbf{T}_i^2)\chi_i + \xi_i \right] / \mathrm{E}\mathbf{T}_i}{1 - \rho}. \qquad (3.48)$$

Note that Φ, and hence also $\tilde{F}_1'(1) = s\Phi$, the mean queue length at Q_1 at polling epochs, do not depend on the service disciplines at the other queues and only depend on the individual parameters of the other queues through ρ and s.

For exhaustive service, i.e., $\mathrm{E}\mathbf{T}_i = \beta_i/(1-\rho_i)$, $\mathrm{E}(\mathbf{T}_i^2) = \beta_i^{(2)}/(1-\rho_i)^3$, $\tau_{ij} = -\lambda_i\lambda_j\mathrm{E}(\mathbf{T}_i^2)$, the expressions for χ_i and ξ_i, $i = 1,\ldots,n$, reduce to those obtained by Srinivasan et al. [169].

For constant switch-over times the last term in (3.41) drops out, so that according to formulas (3.31), (3.38)-(3.41), and (3.47) $\mathrm{E}\mathbf{W}_1 - \mathrm{E}\mathbf{W}_1^0 = (1 + \tilde{h}_1'(1))\mathrm{E}\mathbf{C}\beta_1/(2\mathrm{E}\mathbf{T}_1)$, which illustrates the fact that the relationship between the waiting times in the models with and without switch-over times then takes a remarkably simple form, cf. (3.37). In particular, for exhaustive and gated service $\mathrm{E}\mathbf{W}_1 - \mathrm{E}\mathbf{W}_1^0 = (1-\rho_1)\mathrm{E}\mathbf{C}/2$ and $\mathrm{E}\mathbf{W}_1 - \mathrm{E}\mathbf{W}_1^0 = (1+\rho_1)\mathrm{E}\mathbf{C}/2$, respectively, as obtained in [101] and [79] by using different techniques.

The bulk of the computational effort is involved with the calculation of the quantities χ_i and ξ_i, $i = 1,\ldots,n$, which may be done completely *independent* of the switch-over times. The coefficients $\phi_i^{(k)}$, $i = 1,\ldots,n$, $k = 1,\ldots,\infty$, needed for these calculations, may be computed recursively from (3.43), supplemented with $\phi_1^{(0)} = 1$, $\phi_i^{(0)} = 0$, $i = 2,\ldots,n$. The number of elementary operations (additions, multiplications) involved is $O(n\log_\rho(\epsilon))$ with ϵ the level of accuracy desired. Once the quantities χ_i and ξ_i, $i = 1,\ldots,n$, have been calculated, $\mathrm{E}\mathbf{W}_1$ may be computed for any value of the switch-over times in $O(n)$ elementary operations.

Chapter 4

A pseudo-conservation law for a polling system with a dormant server

4.1 Introduction

In the present chapter we consider a polling system with a *dormant* server, i.e., a polling system in which the server may be allowed to make a halt at a queue when there are no customers present in the system. In the polling literature the server is usually assumed never to idle, in other words, to be switching when not working. In particular the server is assumed to be switching when there are no customers present in the system. As a rare exception, Eisenberg [83] considers a two-queue model with either alternating priority (the exhaustive service discipline at both queues) or strict priority, in which the server remains idling at a queue when there are no customers present in the system. Eisenberg [84] studies a model with an arbitrary number of queues and the exhaustive service discipline at all queues, in which the server does *not* idle. In a recent study [86], Eisenberg shows however how an adapted version of the solution method in [84] may be used to analyze a model in which the server makes a halt at some of the queues when the system is empty. The outline of the solution method in [86] may also be used to treat a similar model with the gated service discipline. Gersht & Marbukh [109] consider a two-queue model with two types of disciplines for switching from one queue to another. For both types of disciplines they show that for some region of the system parameters the discipline that minimizes the mean waiting cost inserts forced idle periods. Liu, Nain, & Towsley [145] identify polling policies, allowing idling as a possible action, that stochastically minimize the total amount of work in the system at an arbitrary epoch. They show that optimal policies are exhaustive and greedy, i.e., the server should neither switch nor idle when at a non-empty queue. In addi-

tion they prove that in symmetric polling systems patient policies are optimal, i.e., when the entire system is empty the server should remain idling at the last visited queue. Gupta & Srinivasan [115] derive explicit expressions for the waiting-time distribution in a similar model as in [86] by using an approach based on the concept of "descendant sets". They show that while a patient server policy is generally better in the sense of a reduction of the amount of work in the system, there do exist cases where a roving server strategy is better. Blanc & Van der Mei [24] use the power-series algorithm to analyze the performance of a system in which the server may be allowed to make a halt at a queue when the entire system is empty. They find that the performance may improve considerably by allowing the server to make a halt at a queue, especially in light traffic.

One reason why usually in the polling literature the server is nevertheless assumed never to idle, may be that the option of idling in general slightly complicates the operation of the system. If at all technically feasible, some mechanism is needed to control the server and to keep track of the customers present in the system. Consequently, the option of idling in general also slightly complicates the mathematical analysis of the system. Another reason may be that the option of idling will have the biggest impact in light traffic, when the performance will be satisfactory anyhow.

However, quite often there are very sound reasons for letting the server stop switching when there are no customers present in the system. In many situations some mechanism to control the server and to keep track of the customers present in the system is needed anyhow. The option of idling then arises quite naturally. In manufacturing and maintenance environments, e.g., one usually requires already some kind of supporting system to schedule the jobs. In such situations it makes sense to let the server make a halt at a queue when the entire system is empty, rather than to let the server needlessly circle around. One option is then allowing the server to make a halt at all of the queues, i.e., to stop switching as soon as the entire system is empty. Another option is allowing the server only to make a halt at some of the queues (thus possibly forcing the server to keep switching for a while), e.g. at a queue which functions as home base or at the queue where a new customer is most likely to arrive. The latter option may be recognized in the dynamic control of traffic lights. When there are no vehicles waiting, typically the main stream is given passage, until a waiting vehicle of a crossing stream is detected.

In many situations there are moreover significant cost involved in switching. In manufacturing and maintenance applications the switch-over usually represents the change-over from one type of jobs to another, which may involve labor cost, material cost, or transportation cost. In such situations a potential saving in switching cost is an additional reason for letting the server stop switching when there are no customers present in the system.

Apart from the practical relevance, it is theoretically interesting to gain some insight into the effect of idling. In the present chapter we therefore derive a pseudo-conservation law for a general model, permitting a wide variety of

service disciplines, in which the server may be allowed to make a halt at an arbitrary subset of queues. A pseudo-conservation law provides a relatively simple expression for a specific weighted sum of the mean waiting times, cf. Section 2.3. By its comparative simplicity a pseudo-conservation law is likely to provide some insight into the effect of idling, whereas the individual mean waiting times themselves involve expressions far too complicated to do so. Linked up with this, the determination of the individual mean waiting times, if at all possible, requires an intricate analysis and relies substantially on the features of the model under consideration, whereas the derivation of a pseudo-conservation law in fact only demands the calculation of mean working/idling times and only marginally leans on the characteristics of the model under consideration.
The remainder of the chapter is organized as follows. In Section 4.2 we present a detailed model description. In Section 4.3 we derive a pseudo-conservation law for the model under consideration. We use the pseudo-conservation law in Section 4.4 to compare the dormant and the non-dormant server case. Further we then address the question at which queues the server should make a halt to minimize the mean total amount of work in the system.

4.2 Model description

The model under consideration consists of n queues, $Q_1, \ldots, Q_n$, each of infinite capacity, attended by a single server S. For the specification of the arrival, service, and switch-over processes we refer to the description of the 'basic model' in Section 1.3.
The server visits the queues in strictly cyclic order, $Q_1, \ldots, Q_n$. As soon as the server arrives at Q_i, it starts serving type-i customers (possibly none), as prescribed by the service discipline. For now we do not specify the service disciplines at the various queues. We only demand the service disciplines at the various queues to be non-preemptive and work-conserving, i.e., no work is created or destroyed, cf. Boxma [38]. As soon as the server has finished serving type-i customers (possibly none), as prescribed by the service discipline, it departs from Q_i. However, we put in the proviso that the server may be allowed to make a halt at Q_i when there are no customers present in the system. The server then remains idling at Q_i, awaiting a new customer to arrive at one of the queues. For now we do not specify the additional criteria for deciding when to make a halt. In case of the 1-limited service discipline e.g., it seems to make sense only to make a halt at a queue when the server has not served any customer yet. As soon as a new customer arrives, the server resumes its activities. If the new customer arrives at Q_i and if the service discipline permits to do so, then the server immediately starts serving the newly arrived customer. Otherwise the server immediately starts switching to Q_{i+1}. The newly arrived customer then does not need to be served first, as, on its way to the newly arrived customer, the server may encounter other customers at other queues.
The case in which the server makes a halt at none of the queues when there are

no customers present in the system, will also be referred to as the non-dormant server case. The case in which the server makes a halt at all of the queues when there are no customers present in the system, will also be referred to as the purely dormant server case.
As a particular variant of a polling system with a dormant server, we focus in the next chapter on a globally gated polling system, in which the server only makes a halt at its home base. In Eisenberg [86] the latter stopping convention is referred to as the "Continue-Cycle-To-Home-Base" rule, as opposed to the "Jump-Directly-To-Home-Base" rule, where the server, on emptying the system, executes a single change-over that takes it directly to the home base. In the purely dormant server case Eisenberg [86] speaks of the "Stop Immediately" rule. Analogously, the starting convention that we adopted here is referred to as the "Resume-Cycle" rule, as opposed to the "Jump-Directly-To-New-Arrival" rule, where the server executes a single change-over that takes it directly to the queue receiving the new arrival.
Finally some words on the stability conditions. We claim - without formal proof - that the opportunity to idle when the system is empty does not affect the stability conditions as discussed for the ordinary non-dormant server case in Section 1.3. The reason is that if the stability conditions were violated, then the system would be empty with probability 0, so that also the opportunity to idle would arise only with probability 0. Throughout the chapter the stability conditions are assumed to hold.

4.3 A PSEUDO-CONSERVATION LAW

In this section we derive a pseudo-conservation law for the model under consideration. In the next section we will use the pseudo-conservation law to compare the dormant and the non-dormant server case.
We first introduce some notation. Denote by π_i the probability that at an arbitrary epoch the server is idling at Q_i, $i = 1, \ldots, n$. If the server is not allowed to make a halt at Q_i, then of course $\pi_i \equiv 0$. In general the probabilities π_i are not simply known. For the exhaustive and gated service discipline they can be determined along the lines of [86]. For the Bernoulli service discipline (comprising the exhaustive and 1-limited service discipline as extreme cases), they can be determined numerically using the power-series algorithm, cf. [24]. The total probability that at an arbitrary epoch the server is idling is $\pi := \sum_{i=1}^{n} \pi_i$.
Denote by $\mathbf{C}_i$ the cycle time with respect to Q_i, i.e., the time between two successive polling epochs at Q_i, $i = 1, \ldots, n$. Although the distribution of $\mathbf{C}_i$ in general depends on i, $\mathrm{E}\mathbf{C}_i$ obviously does not. Noting that the server is working a fraction ρ of the time and idling a fraction π of the time,

$$\mathrm{E}\mathbf{C}_i = \frac{s}{1-\rho-\pi}, \qquad i = 1, \ldots, n. \tag{4.1}$$

Denote by $\mathbf{V}_i$ the total time that the server is working at Q_i during a cycle, $i = 1, \ldots, n$. As ρ_i is the fraction of time that the server is working at Q_i, using (4.1),

$$\mathrm{E}\mathbf{V}_i = \frac{\rho_i s}{1-\rho-\pi}, \qquad i = 1, \ldots, n. \tag{4.2}$$

Denote by $\mathbf{I}_i$ the total time that the server is idling at Q_i during a cycle, $i = 1, \ldots, n$. As π_i is the fraction of time that the server is idling at Q_i, using (4.1),

$$\mathrm{E}\mathbf{I}_i = \frac{\pi_i s}{1-\rho-\pi}, \qquad i = 1, \ldots, n. \tag{4.3}$$

We now derive the pseudo-conservation law for the model under consideration. The approach is similar to the approach in Boxma & Groenendijk [42] for the ordinary non-dormant server case. It is easily verified that the model under consideration satisfies the assumptions mentioned in Boxma [38]. Hence, as described in Section 2.3, the mean waiting times in the model under consideration satisfy the following relationship:

$$\sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i = \rho \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}}{2(1-\rho)} + \mathrm{E}\mathbf{Y}, \tag{4.4}$$

with $\mathbf{Y}$ denoting the amount of work in the system at an arbitrary epoch in a non-serving interval, i.e., a switching interval or an idling interval. We now determine $\mathrm{E}\mathbf{Y}$, by distinguishing whether the server is switching or idling and conditioning on the type of switching interval. Denote by $\mathbf{Y}_i$ the total amount of work in the system at an arbitrary epoch in a switching interval from Q_i to Q_{i+1}, $i = 1, \ldots, n$. By definition there are no customers present in the system when the server is idling, in other words, the total amount of work in the system at an arbitrary epoch in an idling interval is zero. Hence, with $\mathrm{E}\mathbf{I} := \sum_{i=1}^{n} \mathrm{E}\mathbf{I}_i$,

$$\mathrm{E}\mathbf{Y} = \sum_{i=1}^{n} \frac{s_i}{s + \mathrm{E}\mathbf{I}} \mathrm{E}\mathbf{Y}_i. \tag{4.5}$$

$\mathbf{Y}_i$ is composed of two terms, viz.:
i. $\mathbf{Z}_i$, the total amount of work in the system at the beginning of the switch-over from Q_i to Q_{i+1};
ii. the total amount of work that arrives in the system between the beginning of the switch-over from Q_i to Q_{i+1} and the epoch under consideration;
so

$$\mathrm{E}\mathbf{Y}_i = \mathrm{E}\mathbf{Z}_i + \rho \frac{s_i^{(2)}}{2s_i}, \qquad i = 1, \ldots, n. \tag{4.6}$$

We now fragment $\mathbf{Z}_i$ further into work that accumulated at different queues during different periods. Denote by $\mathbf{Z}_{ij}$ the amount of work at Q_j at the beginning of a switch-over from Q_i to Q_{i+1}, $i = 1, \ldots, n$, $j = 1, \ldots, n$. Then

$$\mathrm{E}\mathbf{Z}_i = \sum_{j=1}^{n} \mathrm{E}\mathbf{Z}_{ij}, \qquad i = 1, \ldots, n. \tag{4.7}$$

$\mathbf{Z}_{ij}$ is composed of two terms, viz.:
i. $\mathbf{Z}_{jj}$, the amount of work at Q_j at the beginning of the switch-over from Q_j to Q_{j+1};
ii. the amount of work that arrives at Q_j between the beginning of the switch-over from Q_j to Q_{j+1} and the beginning of the switch-over from Q_i to Q_{i+1};
so, using (4.2) and (4.3),

$$\mathrm{E}\mathbf{Z}_{ij} = \mathrm{E}\mathbf{Z}_{jj} + \rho_j \sum_{k=j+1}^{i} \left(s_{k-1} + \frac{(\rho_k + \pi_k)s}{1-\rho-\pi} \right), \qquad i \neq j, \tag{4.8}$$

where the summation is to be interpreted cyclically.
Substituting (4.6), (4.7), and (4.8) into (4.5), noting that $\dfrac{s}{s + \mathrm{E}\mathbf{I}} = \dfrac{1-\rho-\pi}{1-\rho}$,

$$\begin{aligned}
\mathrm{E}\mathbf{Y} &= \frac{s}{1-\rho} \sum_{i=1}^{n} \sum_{j=1}^{i-1} \rho_i \rho_j + \frac{\rho(1-\rho-\pi)}{(1-\rho)s} \left[\sum_{i=1}^{n} \sum_{j=1}^{i-1} s_i s_j + \frac{1}{2} \sum_{i=1}^{n} s_i^{(2)} \right] + \\
&\quad \frac{1}{1-\rho} \sum_{i=1}^{n} \sum_{j \neq i} \sum_{k=j+1}^{i} s_i \rho_j \pi_k + \frac{1-\rho-\pi}{1-\rho} \sum_{i=1}^{n} \mathrm{E}\mathbf{Z}_{ii} \\
&= \frac{s}{2(1-\rho)} \left[\rho^2 - \sum_{i=1}^{n} \rho_i^2 \right] + \frac{1-\rho-\pi}{1-\rho} \rho \frac{s^{(2)}}{2s} + \\
&\quad \frac{1}{1-\rho} \sum_{i=1}^{n} \sum_{j \neq i} \sum_{k=j+1}^{i} s_i \rho_j \pi_k + \frac{1-\rho-\pi}{1-\rho} \sum_{i=1}^{n} \mathrm{E}\mathbf{Z}_{ii}.
\end{aligned} \tag{4.9}$$

Substituting (4.9) into (4.4), we obtain the following relationship:

Theorem 4.3.1
The mean waiting times in the model under consideration satisfy the following relationship:

$$\sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i = \rho \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}}{2(1-\rho)} + \frac{s}{2(1-\rho)} \left[\rho^2 - \sum_{i=1}^{n} \rho_i^2 \right] + \tag{4.10}$$

$$\frac{1-\rho-\pi}{1-\rho}\rho\frac{s^{(2)}}{2s}+\frac{1}{1-\rho}\sum_{i=1}^{n}\sum_{j\neq i}\sum_{k=j+1}^{i}s_i\rho_j\pi_k+\frac{1-\rho-\pi}{1-\rho}\sum_{i=1}^{n}\mathrm{E}\mathbf{Z}_{ii},$$

with $\mathrm{E}\mathbf{Z}_{ii}$ the mean amount of work that is left behind by the server at Q_i.

We may interpret the terms in (4.10) as follows. The term in the left-hand side is the mean amount of waiting work in the system at an arbitrary epoch. The first term in the right-hand side is the mean amount of waiting work in the corresponding system without switch-over times at an arbitrary epoch. The remaining terms in the right-hand side reflect the influence of the switch-over times. Together these terms constitute the mean total amount of work in the system at an arbitrary epoch in a non-serving interval, i.e., a switching interval or an idling interval. The last term in the right-hand side represents the mean amount of work at an arbitrary epoch in a non-serving interval that is left behind by the server at the various queues. Together the second, third, and fourth term in the right-hand side constitute the mean total amount of work at an arbitrary epoch in a non-serving interval that arrived at the various queues since the server left those queues. Separately the second, third, and fourth term represent the mean amount of work at an arbitrary epoch in a non-serving interval that arrived at the various queues during the working, switching, and idling intervals since the server left those queues, respectively. These terms do not depend on the service disciplines at the various queues, at least as far as their global structure is concerned; the probabilities π_i that occur in these terms probably do depend on the service disciplines at the various queues. Apparently, at least as far as the global structure of (4.10) is concerned, the last term in the right-hand side completely captures the influence of the service disciplines at the various queues.

Remember that we determined the terms in the right-hand side of (4.10), except the first term, by distinguishing whether the server is switching or idling and conditioning on the type of switching interval. The second, third, and fifth term may in fact also be determined without conditioning on the type of switching interval, the fourth term does not allow such an alternative determination.

The last term in the right-hand side of (4.10), representing the mean amount of work that is left behind by the server at the various queues, still remains to be specified. Obviously $\mathrm{E}\mathbf{Z}_{ii}$ is influenced by the service discipline at Q_i. However, as a pleasing circumstance, $\mathrm{E}\mathbf{Z}_{ii}$ is in fact influenced by the service discipline at Q_i *only*, i.e., not by the service discipline at Q_j, $j \neq i$. On the basis of the service discipline at Q_i only, we can split $\mathrm{E}\mathbf{Z}_{ii}$ into work that arrived during working, switching, and idling intervals, whose means do not depend on the service disciplines at the various queues, cf. (4.2) and (4.3).

We now determine $\mathrm{E}\mathbf{Z}_{ii}$ for the exhaustive (I), gated (II), 1-limited (III), and globally gated (IV) service discipline. We need to distinguish whether the server, when idling at Q_i, has already served a customer during its visit to Q_i or not. Denote by π_i' the probability that at an arbitrary epoch the server is

idling at Q_i and has not served any customer yet, $i = 1, \ldots, n$. Denote by π_i'' the probability that at an arbitrary epoch the server is idling at Q_i and has already served a customer, $i = 1, \ldots, n$. Neither π_i' nor π_i'' are simply known, but of course $\pi_i \equiv \pi_i' + \pi_i''$. Let E, G, L, and GG represent the index set of the queues where the exhaustive, gated, 1-limited, and globally gated service discipline is used, respectively.

I. Exhaustive: S serves type-i customers until Q_i is empty. So,

$$\mathrm{E}\mathbf{Z}_{ii} = 0, \quad i \in E. \tag{4.11}$$

II. Gated: S serves exactly those type-i customers present at its arrival at Q_i. A customer that arrives when S is idling at Q_i and has not served any customer yet is also served. So, using (4.2) and (4.3),

$$\mathrm{E}\mathbf{Z}_{ii} = \frac{\rho_i(\rho_i + \pi_i'')s}{1 - \rho - \pi}, \quad i \in G. \tag{4.12}$$

III. 1-Limited: S serves one type-i customer, provided there is a customer present at its arrival at Q_i. A customer that arrives when S is idling at Q_i and has not served any customer yet is also served. Thus S leaves behind the amount of work that arrives during the waiting time of the customer that is possibly served and during its visit to Q_i, but not during the possible idling period before S serves a customer. It follows from (4.1) that on average $\dfrac{\lambda_i s}{1 - \rho - \pi}$ customers are served during a visit to Q_i. In particular, under the 1-limited service discipline, with probability $\dfrac{\lambda_i s}{1 - \rho - \pi}$ a customer is served during a visit to Q_i. So, using (4.2) and (4.3),

$$\mathrm{E}\mathbf{Z}_{ii} = \rho_i \left(\frac{\lambda_i s}{1 - \rho - \pi} \mathrm{E}\mathbf{W}_i + \frac{(\rho_i + \pi_i'')s}{1 - \rho - \pi} \right), \quad i \in L. \tag{4.13}$$

IV. Globally gated: S serves exactly those type-i customers present at its most recent arrival at Q_1. A customer that arrives when S is idling at Q_1 and has not served any customer yet is also served. So, using (4.2) and (4.3),

$$\mathrm{E}\mathbf{Z}_{ii} = \rho_i \left(\frac{(\rho_1 + \pi_1'')s}{1 - \rho - \pi} + \sum_{j=2}^{i} \left(s_{j-1} + \frac{(\rho_j + \pi_j)s}{1 - \rho - \pi} \right) \right), \quad i \in GG. \tag{4.14}$$

We now assume that at each queue either the exhaustive, gated, 1-limited, or globally gated service discipline is used, i.e., $E \cup G \cup L \cup GG = \{1, \ldots, n\}$. Substituting (4.11), (4.12), (4.13), and (4.14) into (4.10), we obtain the following pseudo-conservation law:

Theorem 4.3.2
The mean waiting times in the model under consideration satisfy the following relationship:

$$\sum_{i\in E,G,GG} \rho_i \mathrm{EW}_i + \sum_{i\in L} \rho_i \left(1 - \frac{\lambda_i s}{1-\rho}\right) \mathrm{EW}_i = \tag{4.15}$$

$$\rho \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}}{2(1-\rho)} + \frac{s}{2(1-\rho)} \left[\rho^2 - \sum_{i\in E} \rho_i^2 + \sum_{i\in G,L,GG} \rho_i^2 + 2 \sum_{i\in GG} \sum_{j=1}^{i-1} \rho_i \rho_j\right] +$$

$$\frac{1-\rho-\pi}{1-\rho} \left[\rho \frac{s^{(2)}}{2s} + \sum_{i\in GG} \sum_{j=1}^{i-1} \rho_i s_j\right] +$$

$$\frac{1}{1-\rho} \left[\sum_{i=1}^{n} \sum_{j\neq i} \sum_{k=j+1}^{i} s_i \rho_j \pi_k + s \sum_{i\in G,L} \rho_i \pi_i'' + s \sum_{i\in GG} \rho_i (\pi_1'' + \sum_{j=2}^{i} \pi_j)\right].$$

Elaborating on the explanation of the terms in (4.10), we may interpret the terms in (4.15) as follows. The terms in the left-hand side, excluding the term $\sum_{i\in L} \rho_i \frac{\lambda_i s}{1-\rho} \mathrm{EW}_i$, together constitute the mean amount of waiting work in the system at an arbitrary epoch. The first term in the right-hand side is the mean amount of waiting work in the corresponding system without switch-over times at an arbitrary epoch. The remaining terms in the right-hand side and the term $\sum_{i\in L} \rho_i \frac{\lambda_i s}{1-\rho} \mathrm{EW}_i$ together constitute the mean total amount of work in the system at an arbitrary epoch in a non-serving interval, i.e., a switching interval or an idling interval. Separately the second, third, and fourth term represent the mean amount of work at an arbitrary epoch in a non-serving interval that arrived at the various queues during working, switching, and idling intervals, respectively, excluding the work that arrived at 1-limited queues before the server polled those queues for the last time.

Remark 4.3.1

It is easily verified that the mean waiting times in the model of Eisenberg [83] with the exhaustive service discipline at both queues indeed satisfy (4.15) with $n = 2$, $E = \{1, 2\}$.

For $\pi_i = 0$, $i = 1, \ldots, n$, (4.15) reduces to the pseudo-conservation law for the ordinary non-dormant server case, cf. Boxma & Groenendijk [42].

For $E = \{1, \ldots, n\}$, (4.15) reduces to the pseudo-conservation law obtained by Gupta & Srinivasan [115] for a similar model with the exhaustive service discipline at all queues. In the latter study the pseudo-conservation law is obtained as a by-product from the determination of the individual mean waiting times rather than by using a work decomposition approach.

□

Remark 4.3.2
Notice that the pseudo-conservation law is not as explicit in the dormant server case as in the ordinary non-dormant server case. In the dormant server case, to obtain the pseudo-conservation law explicitly, we are committed to an intricate analysis to determine the probabilities π_i, if at all possible. For the exhaustive and gated service discipline they can be determined along the lines of [86]. For the globally gated service discipline, mathematically a most tractable service discipline, there is no obstacle to determining the probabilities π_i either. For the 1-limited service discipline there is no method available for determining the probabilities π_i analytically. However, they can be determined numerically using the power-series algorithm, cf. [24].

□

Remark 4.3.3
Like in the ordinary non-dormant server case, cf. Groenendijk [113], a pseudo-conservation law is useful for supporting approximations for the mean waiting times and for finding the exact mean waiting times in a symmetric system in a simple manner. Various approximations for the mean waiting times are conceivable, but we do not pursue this matter here any further.

□

Remark 4.3.4
In this section we obtained a pseudo-conservation law for a model with cyclic polling and single Poisson arrivals. Without seriously complicating the above analysis, cyclic polling may be generalized to polling guided by a table, cf. [45], or Markovian polling, i.e., the server visits the queues guided by a Markov chain with state space $\{1, \ldots, n\}$, cf. [53]; and single Poisson arrivals may be generalized to batch Poisson arrivals, cf. Boxma [38].

□

4.4 A COMPARISON BETWEEN THE DORMANT AND THE NON-DORMANT SERVER CASE

In the previous section we obtained a pseudo-conservation law for the model under consideration. In this section we use the pseudo-conservation law to compare the dormant and the non-dormant server case. Specifically we compare the mean waiting times in a symmetric system in the purely dormant and the non-dormant server case. Further we address the question at which queues the server should make a halt to minimize the mean total amount of work in the system.
Let us label the waiting times in the dormant and the non-dormant server case with a hat and a tilde respectively. From (4.15),

$$\sum_{i\in E,G,GG} \rho_i(\mathrm{E}\hat{\mathbf{W}}_i - \mathrm{E}\tilde{\mathbf{W}}_i) + \sum_{i\in L} \rho_i \left(1 - \frac{\lambda_i s}{1-\rho}\right)(\mathrm{E}\hat{\mathbf{W}}_i - \mathrm{E}\tilde{\mathbf{W}}_i) = \quad (4.16)$$

$$-\frac{\pi}{1-\rho}\left[\rho\frac{s^{(2)}}{2s} + \sum_{i\in GG}\sum_{j=1}^{i-1}\rho_i s_j\right] +$$

$$\frac{1}{1-\rho}\left[\sum_{i=1}^{n}\sum_{j\neq i}\sum_{k=j+1}^{i} s_i\rho_j\pi_k + s\sum_{i\in G,L}\rho_i\pi_i'' + s\sum_{i\in GG}\rho_i(\pi_1'' + \sum_{j=2}^{i}\pi_j)\right] =$$

$$-\frac{\rho\pi}{1-\rho}\left\{\left[\frac{s^{(2)}}{2s} + \sum_{i\in GG}\sum_{j=1}^{i-1}\frac{\rho_i}{\rho}s_j\right] -\right.$$

$$\left.\left[\sum_{i=1}^{n}\sum_{j\neq i}\sum_{k=j+1}^{i} s_i\frac{\rho_j}{\rho}\frac{\pi_k}{\pi} + s\sum_{i\in G,L}\frac{\rho_i}{\rho}\frac{\pi_i''}{\pi} + s\sum_{i\in GG}\frac{\rho_i}{\rho}(\frac{\pi_1''}{\pi} + \sum_{j=2}^{i}\frac{\pi_j}{\pi})\right]\right\},$$

where π of course refers to the dormant server case. The interpretation of the terms in the left-hand side and in the first form of the right-hand side of (4.16) follows immediately from the interpretation of the terms in (4.15). The terms in the left-hand side, excluding the term $\sum_{i\in L}\rho_i\frac{\lambda_i s}{1-\rho}(\mathrm{E}\hat{\mathbf{W}}_i - \mathrm{E}\tilde{\mathbf{W}}_i)$, together constitute the difference in the mean total amount of work in the system at an arbitrary epoch between the dormant and the non-dormant server case.

The first term in the first form of the right-hand side represents the difference in the mean amount of work at an arbitrary epoch in a non-serving interval that arrived during *switching* intervals (excluding the work that arrived at 1-limited queues during switching intervals before the server polled those queues for the last time). The first term itself is the product of two terms. The term inside the square brackets represents the mean amount of work at an arbitrary epoch in a switching interval that arrived during switching intervals (excluding the work that arrived at 1-limited queues during switching intervals before the server polled those queues for the last time). Notice that these quantities are the same in the dormant and the non-dormant server case. The term in front is the probability that in the dormant server case an arbitrary epoch in a non-serving interval concerns an idling epoch rather than a switching epoch. The first term is the product of these two terms, as the amount of work at an arbitrary idling epoch in the dormant server case is by definition zero.

The second term in the first form of the right-hand side represents the difference in the mean amount of work at an arbitrary epoch in a non-serving interval that arrived during *idling* intervals. The second term is of course just the mean amount of work at an arbitrary epoch in a non-serving interval that arrived during idling intervals in the dormant server case, as in the non-dormant server

case there are by definition no idling intervals.
The interpretation of the terms in the second form of the right-hand side of (4.16) requires a somewhat different point of view. Define $f(i,j,k)$, $g(i,j,k)$, and $h(i,j,k)$ as the number of times the server needs to switch from Q_i to Q_{i+1} when currently in an empty system switching from Q_k to Q_{k+1}, idling at Q_k not having served any customer yet, and idling at Q_k having already served a customer, respectively, before it can perform work currently arriving at Q_j, *taking into account the service discipline at* Q_j. Then the first and the second term in the first form of the right-hand side may be written as

$$\frac{\pi}{1-\rho}\sum_{i=1}^{n}\frac{s_i}{s}\sum_{j=1}^{n}\rho_j\left(\frac{s_i^{(2)}}{2s_i}+\sum_{k=1}^{n}s_k f(i,j,k)\right)=$$

$$\frac{\pi\rho}{1-\rho}\sum_{i=1}^{n}\frac{s_i}{s}\sum_{j=1}^{n}\frac{\rho_j}{\rho}\left(\frac{s_i^{(2)}}{2s_i}+\sum_{k=1}^{n}s_k f(k,j,i)\right)$$

and

$$\frac{1-\rho-\pi}{1-\rho}\sum_{i=1}^{n}\frac{s_i}{s}\sum_{j=1}^{n}\rho_j\sum_{k=1}^{n}\left(\frac{\pi_k' s}{1-\rho-\pi}g(i,j,k)+\frac{\pi_k'' s}{1-\rho-\pi}h(i,j,k)\right)=$$

$$\frac{\pi\rho}{1-\rho}\left(\sum_{i=1}^{n}\frac{\pi_i'}{\pi}\sum_{j=1}^{n}\frac{\rho_j}{\rho}\sum_{k=1}^{n}s_k g(k,j,i)+\sum_{i=1}^{n}\frac{\pi_i''}{\pi}\sum_{j=1}^{n}\frac{\rho_j}{\rho}\sum_{k=1}^{n}s_k h(k,j,i)\right),$$

respectively. As the amount of work arriving at Q_j constitutes a fraction $\frac{\rho_j}{\rho}$ of the total amount of work arriving to the system, we may interpret

$$\sum_{i=1}^{n}\frac{s_i}{s}\sum_{j=1}^{n}\frac{\rho_j}{\rho}\left(\frac{s_i^{(2)}}{2s_i}+\sum_{k=1}^{n}s_k f(k,j,i)\right)$$

and

$$\sum_{i=1}^{n}\frac{\pi_i'}{\pi}\sum_{j=1}^{n}\frac{\rho_j}{\rho}\sum_{k=1}^{n}s_k g(k,j,i)+\sum_{i=1}^{n}\frac{\pi_i''}{\pi}\sum_{j=1}^{n}\frac{\rho_j}{\rho}\sum_{k=1}^{n}s_k h(k,j,i)$$

as the mean total switch-over time to be incurred by the server when currently in an empty system switching and idling, respectively, before it can perform work currently arriving to the the system (at an arbitrary place), *taking into account the various service disciplines.* For brevity let us refer to these quantities as the mean access time of a switching and an idling server, respectively. Concluding, the difference in the mean total amount of work between the dormant and the non-dormant server case is just the difference in the mean access time between a switching and an idling server, preceded by the multiplier $\frac{\pi\rho}{1-\rho}$.

We now compare the mean waiting times in a symmetric system in the purely dormant and the non-dormant server case for the exhaustive (I), gated (II), and 1-limited (III) service discipline. To do so we take in (4.16) $\rho_i = \rho/n$, $s_i = s/n$, $\pi_i = \pi/n$, $\mathrm{E}\hat{\mathbf{W}}_i = \mathrm{E}\hat{\mathbf{W}}$, $\mathrm{E}\tilde{\mathbf{W}}_i = \mathrm{E}\tilde{\mathbf{W}}$ and for the 1-limited service discipline in addition $\lambda_i = \lambda/n$, $\pi_i'' = \pi''/n$.
We can not compare the mean waiting times in a symmetric system for the globally gated service discipline. Even when the Q_i are all identical, the asymmetric nature of the globally gated service discipline will preclude that the mean waiting times $\mathrm{E}\mathbf{W}_i$ and the probabilities π_i are all identical.
I. Exhaustive:

$$\begin{aligned} \mathrm{E}\hat{\mathbf{W}} - \mathrm{E}\tilde{\mathbf{W}} &= \frac{\pi s}{2(1-\rho)}\left[\frac{n-1}{n} - \frac{s^{(2)}}{s^2}\right] \\ &\leq -\frac{\pi s}{2n(1-\rho)} \leq 0. \end{aligned} \tag{4.17}$$

II. Gated:

$$\begin{aligned} \mathrm{E}\hat{\mathbf{W}} - \mathrm{E}\tilde{\mathbf{W}} &= \frac{s}{2(1-\rho)}\left[\pi\left(\frac{n-1}{n} - \frac{s^{(2)}}{s^2}\right) + \frac{2\pi''}{n}\right] \\ &\leq \frac{\pi s}{2n(1-\rho)} \leq \frac{s}{2n}. \end{aligned} \tag{4.18}$$

III. 1-Limited:

$$\begin{aligned} \mathrm{E}\hat{\mathbf{W}} - \mathrm{E}\tilde{\mathbf{W}} &= \frac{s}{2(1-\rho-\frac{\lambda s}{n})}\left[\pi\left(\frac{n-1}{n} - \frac{s^{(2)}}{s^2}\right) + \frac{2\pi''}{n}\right] \\ &\leq \frac{\pi s}{2(n(1-\rho)-\lambda s)} \leq \frac{s}{2n}. \end{aligned} \tag{4.19}$$

For the exhaustive service discipline always $\mathrm{E}\hat{\mathbf{W}} \leq \mathrm{E}\tilde{\mathbf{W}}$, which agrees with the result of Liu, Nain, & Towsley [145] that in a symmetric system the server should remain idling at a queue when the entire system is empty to minimize the total amount of work. Also $\mathrm{E}\hat{\mathbf{W}} \leq \mathrm{E}\tilde{\mathbf{W}}$ for the gated service discipline when the server is only allowed to make a halt at a queue when it has not served any customer yet, i.e., when $\pi'' = 0$. When the server is also allowed to make a halt at a queue when it has already served a customer, i.e., when $\pi'' = \pi$, $\mathrm{E}\hat{\mathbf{W}} \leq \mathrm{E}\tilde{\mathbf{W}}$ ($\mathrm{E}\hat{\mathbf{W}} \geq \mathrm{E}\tilde{\mathbf{W}}$) for the gated service discipline, iff the coefficient of variation of the total switch-over time is larger (smaller) than $1 + 1/n$. In particular $\mathrm{E}\hat{\mathbf{W}} = \mathrm{E}\tilde{\mathbf{W}}$ when the total switch-over time is Erlang-n distributed, so when the individual switch-over times are exponentially distributed. Because of the memoryless property of the exponential distribution customers then indeed do not observe any difference between the dormant and the non-dormant server case. Similar remarks hold for the 1-limited service discipline.

We finally address the question at which queues the server should make a halt to minimize the mean total amount of work in the system. Liu, Nain, &

Towsley [145] identify polling policies, allowing idling as a possible action, that stochastically minimize the total amount of work in the system at an arbitrary epoch. They show that optimal policies are exhaustive and greedy, i.e., the server should neither switch nor idle when at a non-empty queue. In addition they prove that in symmetric polling systems patient policies are optimal, i.e., when the entire system is empty the server should remain idling at the last visited queue. As a non-exhaustive policy can not minimize the total amount of work, we assume in the sequel the service discipline to be exhaustive.
In asymmetric systems the total amount of work is not always minimal in the purely dormant server case. In some asymmetric systems the total amount of work is even in the non-dormant server case smaller. Consider e.g. a system with $n = 2$, $\lambda_1 = \infty$, $\beta_1 = 0$, $s_1^{(2)} = s_1^2$, $s_2 = 0$. As far as the amount of work is concerned, such a system corresponds in the purely dormant and the non-dormant server case to an ordinary $M/G/1$ queue with set-up times and multiple vacations of length s_1, respectively. In the latter case the amount of work is of course smaller. However, in not a single system the total amount of work is minimal in the non-dormant server case, as we will prove now. If the server makes a halt only at Q_h, then (4.16) reduces to

$$\sum_{i=1}^{n} \rho_i (\mathrm{E}\hat{\mathbf{W}}_i - \mathrm{E}\tilde{\mathbf{W}}_i) = -\frac{\pi_h}{1-\rho} \left[\rho \frac{s^{(2)}}{2s} - \sum_{i \neq h-1} \sum_{j=i+1}^{h-1} s_i \rho_j \right].$$

So it suffices to show that there is at least one Q_h such that $\sum_{i \neq h-1} \sum_{j=i+1}^{h-1} s_i \rho_j \leq \rho \frac{s^{(2)}}{2s}$. Now $\sum_{h=1}^{n} \sum_{i \neq h-1} \sum_{j=i+1}^{h-1} s_i \rho_j = \frac{1}{2} n \rho s - \sum_{i=1}^{n} s_i \rho_i$ implies $\min_{1 \leq h \leq n} \sum_{i \neq h-1} \sum_{j=i+1}^{h-1} s_i \rho_j \leq \frac{1}{2} \rho s \leq \rho \frac{s^{(2)}}{2s}$. So there is indeed at least one such Q_h.

We now *know* that making a halt only at Q_h is beneficial, iff $\sum_{i \neq h-1} \sum_{j=i+1}^{h-1} s_i \rho_j \leq \rho \frac{s^{(2)}}{2s}$. In addition we now *assume* that Q_h belongs to the set of queues at which the server should make a halt, when making a halt only at Q_h is beneficial. In other words, we propose to let the server make a halt at Q_h, iff $\sum_{i \neq h-1} \sum_{j=i+1}^{h-1} s_i \rho_j \leq \rho \frac{s^{(2)}}{2s}$. In accordance with the interpretation of the second form of (4.16), the criterion is seen to select queues Q_h, such that the mean access time of the server is smaller when idling at Q_h than when switching. Written as $\sum_{i \neq h-1} \sum_{j=i+1}^{h-1} \frac{s_i}{s} \frac{\rho_j}{\rho} \leq \frac{s^{(2)}}{2s^2}$, the criterion is seen to suggest idling at queues Q_h preceded by queues with relatively light traffic and large switch-over

times and followed by queues with relatively heavy traffic and small switch-over times, and is also seen to suggest idling more strongly accordingly as the variability of the total switch-over time is larger. Notice that in a symmetric system the inequality reduces to $1 - 1/n \leq \frac{s^{(2)}}{s^2}$, which always holds, cf. (4.17).
Blanc & Van der Mei [24] obtain a similar rule from light-traffic considerations. Numerical experiments in [24] suggest that the rule performs very well.

Remark 4.4.1
The problem at which queues the server should make a halt to minimize the mean total amount of work in the system, may be formulated as a semi-Markov decision problem, cf. Tijms [184] p. 200. The decision epochs are the epochs of a visit completion; the possible decisions (actions) are either switching or idling when the visit completion leaves the entire system empty and only switching otherwise. The states are $(i, l_1, \ldots, l_n)$, where i is the queue at which the visit completion occurs and l_j is the number of waiting customers at Q_j, $j = 1, \ldots, n$. The crucial observation is that the system under consideration satisfies the following Markovian property: given the state at some decision epoch and the decision (action) chosen, the evolution after that decision epoch does not depend on the evolution before that decision epoch.
A semi-Markov decision problem formulates the problem of finding a strategy that minimizes the mean total cost per unit of time. A strategy prescribes here at which queues the server should make a halt. The mean total cost per unit of time may be related here to the mean total amount of work in the system, when we define the cost appropriately. If c_i represents the waiting cost per unit of time of an arbitrary type-i customer, $i = 1, \ldots, n$, then the mean total cost per unit of time equal $\sum_{i=1}^{n} c_i \lambda_i \mathrm{E}\mathbf{W}_i$. When we take $c_i = \beta_i$, $i = 1, \ldots, n$, the mean total cost per unit of time equal $\sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i$, the mean amount of waiting work in the system. Minimizing the mean amount of waiting work and minimizing the mean total amount of work are equivalent, as the difference, the mean amount of work in service, always equals $\frac{1}{2} \sum_{i=1}^{n} \lambda_i \beta_i^{(2)}$.
It is straightforward, when action a is chosen in the current state h, to calculate $p(a, h, k)$, the probability that at the next decision epoch the state will be k, $t(a, h)$, the expected time until the next decision epoch, and $c(a, h)$, the expected cost incurred until the next decision epoch. The resulting semi-Markov decision problem may be solved numerically by truncating the state space.
□

Remark 4.4.2
In the present chapter we allowed the server only to make a halt at a queue when there are no customers present in the system. In fact we may allow the server also to make a halt at a queue in other cases when there are customers

present in the system. A first option might be to maintain the service disciplines at the various queues, but to decide at the completion of *each* visit whether to switch or to idle, and not only at the completion of a visit that leaves the entire system empty. A second option might be also to drop the service disciplines at the various queues, and to decide at the completion of each *service* whether to serve another customer if present, to switch, or to idle, like Liu, Nain, & Towsley [145].

Once having enlarged the freedom of decisions in the operation of the system, it is quite natural to consider the problem of finding a strategy that optimizes the performance of the system. As the enlarged freedom of decisions will complicate the analysis even further, there is little hope to solve the problem analytically. Remember that the pseudo-conservation law did not even hold enough information to solve the more specific problem at which queues the server should make a halt to minimize the mean total amount of work in the system. However, like the latter more specific problem, the problem of finding a strategy that minimizes $\sum_{i=1}^{n} c_i \lambda_i \mathrm{E}\mathbf{W}_i$ may still be handled as a semi-Markov decision problem.

□

Chapter 5

A globally gated polling system with a dormant server

5.1 Introduction

In the previous chapter we considered a polling system with a dormant server. We obtained a pseudo-conservation law for a general model, permitting a wide variety of service disciplines, in which the server may be allowed to make a halt at an arbitrary subset of queues. In the present chapter we focus on a particular model from that broad class, in which the option of idling arises quite naturally: a system with the globally gated service discipline, in which the server only makes a halt at its home base. The globally gated service discipline, introduced in Boxma, Levy, & Yechiali [50], operates as follows. Suppose the server arrives at its home base. Then all the customers in the system are marked instantaneously and the server immediately starts a tour along the queues. During this tour exactly the marked customers are served. The service of customers that meanwhile arrive in the system is deferred until the next tour along the queues. Boxma, Weststrate, & Yechiali [41] propose the globally gated service discipline to be used by a repair crew, in charge of the maintenance activities at several installations. As indicated in [41], under the globally gated service discipline it does not make sense to start a tour along the queues when there are no customers present in the system. In the present chapter we therefore consider a globally gated polling system in which the server makes a halt at its home base when there are no customers present in the system. The globally gated service discipline then operates as follows. Suppose again the server arrives at its home base. If there are customers present in the system, they are all marked instantaneously and the server starts a tour along the queues, acting as described before. If there are no customers present in the system, the server remains idling at its home base, awaiting a new customer to

arrive at one of the queues. As soon as a new customer arrives, it is marked instantaneously and the server starts a tour along the queues. During this tour only the newly arrived customer is served. The service of customers that meanwhile arrive in the system is again deferred until the next tour along the queues.
The remainder of the chapter is organized as follows. In Section 5.2 we present a detailed model description. In Section 5.3 we derive an explicit expression for the Laplace-Stieltjes Transform (LST) of the cycle time distribution. We obtain the LST of the waiting time-distribution at each of the queues in Section 5.4. As a justification of the dormant server policy, we show the waiting time at each of the queues to be smaller (in the increasing-convex-ordering sense) than in the ordinary non-dormant server case. In Section 5.5 we derive the probability generating function (pgf) of the joint queue length distribution at polling epochs.

5.2 Model description

The model under consideration consists of n queues, $Q_1, \ldots, Q_n$, each of infinite capacity, attended by a single server S. For the specification of the arrival, service, and switch-over processes we refer to the description of the 'basic model' in Section 1.3.
The server visits the queues in strictly cyclic order, $Q_1, \ldots, Q_n$. Suppose the server is just about to visit Q_1. If there are customers present in the system, they are all marked instantaneously and the server immediately starts visiting $Q_1, \ldots, Q_n$. During the coming cycle exactly the marked customers are served. At each queue customers are served in order of arrival. The service of customers that meanwhile arrive in the system is deferred until the next cycle. If there are no customers present in the system, the server remains idling at Q_1, awaiting a new customer to arrive at one of the queues. As soon as a new customer arrives, it is marked instantaneously and the server starts visiting $Q_1, \ldots, Q_n$. During the coming cycle only the newly arrived customer is served. Again, customers that meanwhile arrive in the system are served during the next cycle. During the cycle the server is not allowed to make a halt at a queue when the completion of a service leaves the system empty. In other words, the server is only allowed to make a halt when the system is empty at the beginning of a visit to Q_1.

Remark 5.2.1
For $n = 1$ the model under consideration reduces to a gated vacation model with *single* vacations, while in the non-dormant server case the model corresponds to a gated vacation model with *multiple* vacations, cf. Takagi [174] pp. 205-213. The latter model is analyzed in detail in Takine & Hasegawa [177].

□

5.3 THE CYCLE TIME

In this section we relate the cycle time distribution to the joint queue length distribution at the beginning of a cycle and at the beginning of a subsequent cycle. The approach is similar to the approach in Boxma, Levy, & Yechiali [50] for the ordinary non-dormant server case. Assuming an equilibrium distribution, we obtain a functional equation for the pgf of the joint queue length distribution at the beginning of a cycle and for the LST of the cycle time distribution. The latter functional equation is solved explicitly. The cycle time distribution will play a crucial role in the analysis of the waiting-time distribution and the joint queue length distribution at polling epochs in Sections 5.4 and 5.5, respectively.

We first introduce some notation. Denote by $\mathbf{C}^{(m)}$ the length of the m-th cycle, i.e., the time between the start of the m-th visit to Q_1 and the start of the $(m+1)$-th visit to Q_1, $m = 1, 2, \ldots$. Denote by $\mathbf{I}^{(m)}$ the length of the m-th idling period, i.e., the m-th idling time at Q_1 (possibly zero), $m = 1, 2, \ldots$. Denote by $\bar{\mathbf{C}}^{(m)}$ the length of the m-th *restricted* cycle, i.e., the m-th cycle time minus the m-th idling time, $m = 1, 2, \ldots$. Let $\alpha_m(\zeta, \omega) := \mathrm{E}(\mathrm{e}^{-\zeta \mathbf{I}^{(m)} - \omega \bar{\mathbf{C}}^{(m)}})$ for $\mathrm{Re}\,\zeta \geq 0$, $\mathrm{Re}\,\omega \geq 0$, $m = 1, 2, \ldots$. Let $\gamma_m(\omega) := \mathrm{E}(\mathrm{e}^{-\omega \bar{\mathbf{C}}^{(m)}})$ for $\mathrm{Re}\,\omega \geq 0$, $m = 1, 2, \ldots$. Denote by $\mathbf{Y}_i^{(m)}$ the number of customers present at queue i at the beginning of the m-th cycle, $i = 1, \ldots, n$, $m = 1, 2, \ldots$. Let $\xi_m(z_1, \ldots, z_n) := \mathrm{E}(z_1^{\mathbf{Y}_1^{(m)}} \ldots z_n^{\mathbf{Y}_n^{(m)}})$ for $|z_i| \leq 1$, $i = 1, \ldots, n$, $m = 1, 2, \ldots$.

By the nature of the globally gated service discipline $\mathbf{I}^{(m)}$, $\bar{\mathbf{C}}^{(m)}$, $\mathbf{Y}_i^{(m)}$, and $\mathbf{Y}_i^{(m+1)}$, $i = 1, \ldots, n$, $m = 1, 2, \ldots$, are related as follows. On the one hand $\mathbf{Y}_i^{(m)}$ equals the number of customers that are served at Q_i during the m-th restricted cycle, unless $(\mathbf{Y}_1^{(m)}, \ldots, \mathbf{Y}_n^{(m)}) = (0, \ldots, 0)$, i.e., there are no customers present at the beginning of the m-th cycle. In that case the server remains idling at Q_1 for a period, which is negative exponentially distributed with parameter λ, until a customer arrives at one of the queues; such an arrival occurs at Q_i with probability λ_i/λ. So

$$\mathrm{E}(\mathrm{e}^{-\zeta \mathbf{I}^{(m)} - \omega \bar{\mathbf{C}}^{(m)}} \mid \mathbf{Y}_1^{(m)}, \ldots, \mathbf{Y}_n^{(m)}) = \tag{5.1}$$

$$\sigma(\omega) \left[\prod_{i=1}^{n} \beta_i(\omega)^{\mathbf{Y}_i^{(m)}} - \left(1 - \frac{\lambda}{\lambda + \zeta} \sum_{i=1}^{n} \frac{\lambda_i}{\lambda} \beta_i(\omega) \right) \mathrm{I}_{\{\mathbf{Y}^{(m)} = 0\}} \right],$$

with $\mathrm{I}_{\{\mathbf{Y}^{(m)} = 0\}}$ denoting the indicator function of the event $(\mathbf{Y}_1^{(m)}, \ldots, \mathbf{Y}_n^{(m)}) = (0, \ldots, 0)$.

Unconditioning (5.1),

$$\alpha_m(\zeta, \omega) = \tag{5.2}$$

$$\sigma(\omega) \left[\xi_m(\beta_1(\omega), \ldots, \beta_n(\omega)) - \left(1 - \frac{\lambda}{\lambda + \zeta} \sum_{i=1}^{n} \frac{\lambda_i}{\lambda} \beta_i(\omega) \right) \xi_m(0, \ldots, 0) \right].$$

On the other hand $\mathbf{Y}_i^{(m+1)}$ equals the number of customers that arrive at Q_i during the m-th restricted cycle. So

$$\mathrm{E}(z_1^{\mathbf{Y}_1^{(m+1)}} \ldots z_n^{\mathbf{Y}_n^{(m+1)}} \mid \bar{\mathbf{C}}^{(m)} = t) = \mathrm{e}^{-\sum_{i=1}^{n} \lambda_i (1-z_i)t}. \tag{5.3}$$

Define $\epsilon(z_1, \ldots, z_n) := \sum_{i=1}^{n} \lambda_i (1 - z_i)$ for $\mid z_i \mid \leq 1$, $i = 1, \ldots, n$.
Unconditioning (5.3),

$$\xi_{m+1}(z_1, \ldots, z_n) = \gamma_m(\epsilon(z_1, \ldots, z_n)). \tag{5.4}$$

Denote by $\mathbf{C}$, $\mathbf{I}$, $\bar{\mathbf{C}}$, and $\mathbf{Y}_i$ stochastic variables with the limiting distribution for $m \to \infty$ of $\mathbf{C}^{(m)}$, $\mathbf{I}^{(m)}$, $\bar{\mathbf{C}}^{(m)}$, and $\mathbf{Y}_i^{(m)}$, $i = 1, \ldots, n$, respectively. Let $\alpha(\zeta, \omega) := \mathrm{E}(\mathrm{e}^{-\zeta \mathbf{I} - \omega \bar{\mathbf{C}}})$ for $\mathrm{Re}\, \zeta \geq 0$, $\mathrm{Re}\, \omega \geq 0$. Let $\gamma(\omega) := \mathrm{E}(\mathrm{e}^{-\omega \bar{\mathbf{C}}})$ for $\mathrm{Re}\, \omega \geq 0$. Let $\xi(z_1, \ldots, z_n) := \mathrm{E}(z_1^{\mathbf{Y}_1}, \ldots, z_n^{\mathbf{Y}_n})$ for $\mid z_i \mid \leq 1$, $i = 1, \ldots, n$.
From (5.2) and (5.4),

$$\alpha(\zeta, \omega) = \tag{5.5}$$

$$\sigma(\omega) \left[\xi(\beta_1(\omega), \ldots, \beta_n(\omega)) - \left(1 - \frac{\lambda}{\lambda + \zeta} \sum_{i=1}^{n} \frac{\lambda_i}{\lambda} \beta_i(\omega) \right) \xi(0, \ldots, 0) \right],$$

and

$$\xi(z_1, \ldots, z_n) = \gamma(\epsilon(z_1, \ldots, z_n)). \tag{5.6}$$

Combining (5.5) and (5.6),

$$\alpha(\zeta, \omega) = \tag{5.7}$$

$$\sigma(\omega) \left[\gamma\left(\sum_{i=1}^{n} \lambda_i (1 - \beta_i(\omega)) \right) - \gamma(\lambda) \left(1 - \frac{\lambda}{\lambda + \zeta} \sum_{i=1}^{n} \frac{\lambda_i}{\lambda} \beta_i(\omega) \right) \right],$$

and

$$\xi(z_1, \ldots, z_n) = \tag{5.8}$$

$$\sigma(\epsilon(z)) \left[\xi(\beta_1(\epsilon(z)), \ldots, \beta_n(\epsilon(z))) - \left(1 - \sum_{i=1}^{n} \frac{\lambda_i}{\lambda} \beta_i(\epsilon(z)) \right) \xi(0, \ldots, 0) \right].$$

Remark 5.3.1
As mentioned in Section 1.4, strictly speaking the globally gated service discipline does not satisfy Property 1.4.1, which provides a global characterization of the class of service disciplines that allow an exact analysis in a multi-type branching process framework. Still the joint queue length process at the beginning of a cycle, $\{(\mathbf{Y}_1^{(m)}, \ldots, \mathbf{Y}_n^{(m)}), m = 1, 2, \ldots\}$, constitutes a multi-type

branching process with state-dependent immigration. The crucial observation is that the globally gated service discipline satisfies the following property: if there are k_i customers present at Q_i at the beginning of a cycle, then each of these k_i customers will be 'effectively replaced' in an i.i.d. manner by a random population having pgf $\beta_i(\epsilon(z))$. Adopting the terminology of the theory of multi-type branching processes, the offspring generating functions are given by $f_i(z) = \beta_i(\epsilon(z))$, $i = 1, \ldots, n$, the immigration generating function for the non-zero states is given by $g(z) = \sigma(\epsilon(z))$, and the immigration generating function for the zero state is given by $g(z)h(z)$ with $h(z) = \sum_{i=1}^{n} \frac{\lambda_i}{\lambda} f_i(z)$. From the theory of multi-type branching processes we have

$$\xi(z_1, \ldots, z_n) = g(z)[\xi(f_1(z), \ldots, f_n(z)) - (1 - h(z))\xi(0, \ldots, 0)],$$

which is identical to (5.8).

□

Below we solve the functional equation (5.7). We first derive some preliminary results from (5.7). Noting that $\mathrm{E}(\mathrm{e}^{-\zeta \mathbf{I}}) = \alpha(\zeta, 0)$ for $\mathrm{Re}\,\zeta \geq 0$ and $\mathrm{E}(\mathrm{e}^{-\omega \bar{\mathbf{C}}}) = \alpha(0, \omega)$ for $\mathrm{Re}\,\omega \geq 0$,

$$\mathrm{E}\mathbf{C} = \frac{s + \frac{\gamma(\lambda)}{\lambda}}{1 - \rho}; \tag{5.9}$$

$$\mathrm{E}\mathbf{I} = \frac{\gamma(\lambda)}{\lambda}; \tag{5.10}$$

$$\mathrm{E}\bar{\mathbf{C}} = \frac{s + \rho \frac{\gamma(\lambda)}{\lambda}}{1 - \rho}; \tag{5.11}$$

$$\mathrm{E}\bar{\mathbf{C}}^2 = \frac{s^{(2)} + (2\rho s + \sum_{i=1}^{n} \lambda_i \beta_i^{(2)})\mathrm{E}\mathbf{C}}{1 - \rho^2}. \tag{5.12}$$

Remark 5.3.2
We may also obtain (5.10) directly by observing

$$\mathrm{E}(\mathbf{I} \mid \mathbf{I} > 0) = \frac{1}{\lambda},$$

while

$$\Pr\{\mathbf{I} > 0\} = \Pr\{(\mathbf{Y}_1, \ldots, \mathbf{Y}_n) = (0, \ldots, 0)\} = \int_{t=0}^{\infty} \mathrm{e}^{-\lambda t} d\Pr\{\bar{\mathbf{C}} < t\} = \gamma(\lambda).$$

We may also obtain (5.9) directly from (4.1) and (5.10) by observing that $\pi = \dfrac{\mathbf{EI}}{\mathbf{EC}}$.

□

We now solve the functional equation (5.7). Obviously it suffices to find an expression for $\gamma(\omega)$, as substituting such an expression into (5.7) yields an expression for $\alpha(\zeta, \omega)$. Define $\delta(\omega) := \sum_{i=1}^{n} \lambda_i(1 - \beta_i(\omega))$ for $\mathrm{Re}\,\omega \geq 0$. Putting $\zeta = 0$ in (5.7),

$$\gamma(\omega) = \sigma(\omega)\left[\gamma(\delta(\omega)) - \frac{\gamma(\lambda)}{\lambda}\delta(\omega)\right], \qquad \mathrm{Re}\,\omega \geq 0. \tag{5.13}$$

Define recursively

$$\begin{aligned} \delta^{(0)}(\omega) &:= \omega, && \mathrm{Re}\,\omega \geq 0; \\ \delta^{(k)}(\omega) &:= \delta(\delta^{(k-1)}(\omega)), && \mathrm{Re}\,\omega \geq 0, k = 1, 2, \ldots . \end{aligned}$$

Iterating (5.13) K times,

$$\gamma(\omega) = \prod_{k=0}^{K} \sigma(\delta^{(k)}(\omega))\gamma(\delta^{(K+1)}(\omega)) - \tag{5.14}$$

$$\frac{\gamma(\lambda)}{\lambda} \sum_{k=0}^{K} \delta^{(k+1)}(\omega) \prod_{l=0}^{k} \sigma(\delta^{(l)}(\omega)), \ \mathrm{Re}\,\omega \geq 0, \ K = 1, 2, \ldots .$$

Lemma 5.3.1

i. $\lim_{K \to \infty} \delta^{(K)}(\omega) = 0$ for all ω with $\mathrm{Re}\,\omega \geq 0$.

ii. $\prod_{k=0}^{\infty} \sigma(\delta^{(k)}(\omega))$ converges for all ω with $\mathrm{Re}\,\omega \geq 0$.

iii. $\sum_{k=0}^{\infty} \delta^{(k+1)}(\omega) \prod_{l=0}^{k} \sigma(\delta^{(l)}(\omega))$ converges for all ω with $\mathrm{Re}\,\omega \geq 0$.

Proof

See Appendix 5.A.

□

Lemma 5.3.1 implies, letting $K \to \infty$ in (5.14),

$$\gamma(\omega) = \prod_{k=0}^{\infty} \sigma(\delta^{(k)}(\omega)) - \frac{\gamma(\lambda)}{\lambda} \sum_{k=0}^{\infty} \delta^{(k+1)}(\omega) \prod_{l=0}^{k} \sigma(\delta^{(l)}(\omega)). \tag{5.15}$$

Putting $\omega = \lambda$ in (5.15),

$$\gamma(\lambda) = \frac{\prod_{k=0}^{\infty} \sigma(\delta^{(k)}(\lambda))}{1 + \frac{1}{\lambda} \sum_{k=0}^{\infty} \delta^{(k+1)}(\lambda) \prod_{l=0}^{k} \sigma(\delta^{(l)}(\lambda))}. \tag{5.16}$$

Substituting (5.16) back into (5.15),

$$\gamma(\omega) = \prod_{k=0}^{\infty} \sigma(\delta^{(k)}(\omega)) - \tag{5.17}$$

$$\frac{\prod_{k=0}^{\infty} \sigma(\delta^{(k)}(\lambda))}{\lambda + \sum_{k=0}^{\infty} \delta^{(k+1)}(\lambda) \prod_{l=0}^{k} \sigma(\delta^{(l)}(\lambda))} \sum_{k=0}^{\infty} \delta^{(k+1)}(\omega) \prod_{l=0}^{k} \sigma(\delta^{(l)}(\omega)).$$

Substituting (5.17) into (5.7) yields an expression for $\alpha(\zeta, \omega)$.
Finally some words on the joint past and residual lifetime distribution of a restricted cycle. This distribution will play a crucial role in the analysis of the waiting-time distribution in the next section. Denote by $\bar{\mathbf{C}}_P$ and $\bar{\mathbf{C}}_R$ stochastic variables with the distribution of the past and residual lifetime of a restricted cycle, respectively. From Cohen [73] p. 113 we have,

$$\mathrm{E}(\mathrm{e}^{-\omega_P \bar{\mathbf{C}}_P - \omega_R \bar{\mathbf{C}}_R}) = \frac{1}{\mathrm{E}\bar{\mathbf{C}}} \frac{\gamma(\omega_R) - \gamma(\omega_P)}{\omega_P - \omega_R}, \quad \mathrm{Re}\,\omega_P \geq 0, \mathrm{Re}\,\omega_R \geq 0, \tag{5.18}$$

In particular,

$$\mathrm{E}(\mathrm{e}^{-\omega \bar{\mathbf{C}}_P}) = \mathrm{E}(\mathrm{e}^{-\omega \bar{\mathbf{C}}_R}) = \frac{1 - \gamma(\omega)}{\omega \mathrm{E}\bar{\mathbf{C}}}, \qquad \mathrm{Re}\,\omega \geq 0. \tag{5.19}$$

From (5.19),

$$\mathrm{E}\bar{\mathbf{C}}_P = \mathrm{E}\bar{\mathbf{C}}_R = \frac{\mathrm{E}\bar{\mathbf{C}}^2}{2\mathrm{E}\bar{\mathbf{C}}}. \tag{5.20}$$

5.4 The waiting time

In formula (5.18) in the previous section we expressed the LST of the joint past and residual lifetime distribution of a restricted cycle into the LST of the restricted cycle time distribution. In this section we relate the waiting-time distribution to the joint past and residual lifetime distribution of a restricted cycle. Thus we obtain an expression for the LST of the waiting-time distribution in terms of the LST of the restricted cycle time distribution.
We first introduce some notation. Denote by $\mathbf{V}_i(\mathbf{N})$ the total service time of $\mathbf{N}$

type-i customers, $i = 1, \ldots, n$, for any non-negative integer valued stochastic variable $\mathbf{N}$. So $\mathrm{E}(\mathrm{e}^{-\omega \mathbf{V}_i(\mathbf{N})}) = \mathrm{E}(\beta_i(\omega)^{\mathbf{N}})$, $\mathrm{Re}\,\omega \geq 0$, $i = 1, \ldots, n$. Denote by $\mathbf{A}_i(\mathbf{T})$ the number of type-i customers arriving during a period of length $\mathbf{T}$, $i = 1, \ldots, n$, for any non-negative real valued stochastic variable $\mathbf{T}$. So $\mathrm{E}(z_i^{\mathbf{A}_i(\mathbf{T})}) = \mathrm{E}(\mathrm{e}^{-\lambda_i(1-z_i)\mathbf{T}})$, $\mid z_i \mid \leq 1$, $i = 1, \ldots, n$. Denote by $\mathbf{B}_i$ and $\mathbf{S}_i$ stochastic variables having distribution $B_i(\cdot)$ and $S_i(\cdot)$, $i = 1, \ldots, n$, respectively.

We now analyze the waiting-time distribution of an arbitrary type-i customer, by distinguishing whether the customer arrives during a restricted cycle or during an idling period (thus terminating the idling period immediately by initiating a new restricted cycle), in other words, whether the customer sees the server working/switching or idling upon arrival. The waiting time $\mathbf{W}_i^{(\mathrm{B})}$ of an arbitrary type-i customer that arrives during a restricted cycle, is composed of

i. the residual lifetime of the restricted cycle in which it arrives;

ii. the total service time of all customers that arrive at $Q_1, \ldots, Q_{i-1}$ during the same restricted cycle;

iii. the total service time of all customers that arrive at Q_i during the past lifetime of the restricted cycle in which it arrives;

iv. the total switch-over time incurred by the server when moving from Q_1 to Q_i;

i.e.,

$$\mathbf{W}_i^{(\mathrm{B})} \stackrel{d}{=} \bar{\mathbf{C}}_R + \sum_{j=1}^{i-1} \mathbf{V}_j(\mathbf{A}_j(\bar{\mathbf{C}}_P + \bar{\mathbf{C}}_R)) + \mathbf{V}_i(\mathbf{A}_i(\bar{\mathbf{C}}_P)) + \sum_{j=1}^{i-1} \mathbf{S}_j.$$

So, using (5.18),

$$\mathrm{E}(\mathrm{e}^{-\omega \mathbf{W}_i^{(\mathrm{B})}}) = \prod_{j=1}^{i-1} \sigma_j(\omega) \times \tag{5.21}$$

$$\int_{t_P=0}^{\infty} \int_{t_R=0}^{\infty} \mathrm{e}^{-\omega t_R} \prod_{j=1}^{i-1} \left\{ \mathrm{e}^{-\lambda_j(1-\beta_j(\omega))(t_P+t_R)} \right\} \mathrm{e}^{-\lambda_i(1-\beta_i(\omega))t_P} \times$$

$$d_{t_P,t_R} \mathrm{Pr}\{\bar{\mathbf{C}}_P < t_P, \bar{\mathbf{C}}_R < t_R\} =$$

$$\prod_{j=1}^{i-1} \sigma_j(\omega) \frac{1}{\mathrm{E}\bar{\mathbf{C}}} \frac{\gamma(\sum_{j=1}^{i} \lambda_j(1-\beta_j(\omega))) - \gamma(\sum_{j=1}^{i-1} \lambda_j(1-\beta_j(\omega)) + \omega)}{\omega - \lambda_i(1-\beta_i(\omega))}.$$

The waiting time $\mathbf{W}_i^{(\mathrm{I})}$ of an arbitrary type-i customer that arrives during an idling period, is composed solely of the total switch-over time incurred by the server when moving from Q_1 to Q_i, i.e.,

$$\mathbf{W}_i^{(\mathrm{I})} \stackrel{d}{=} \sum_{j=1}^{i-1} \mathbf{S}_j.$$

So,

$$\mathrm{E}(\mathrm{e}^{-\omega \mathbf{W}_i^{(\mathrm{I})}}) = \prod_{j=1}^{i-1} \sigma_j(\omega). \tag{5.22}$$

Combining (5.21) and (5.22), noting that an arbitrary customer, irrespective of which type, arrives during a restricted cycle and an idling period with probability $\mathrm{E}\bar{\mathbf{C}}/\mathrm{EC}$ and EI/EC, respectively,

$$\mathrm{E}(\mathrm{e}^{-\omega \mathbf{W}_i}) = \tag{5.23}$$

$$\prod_{j=1}^{i-1} \sigma_j(\omega) \frac{1}{\mathrm{EC}} \left[\mathrm{EI} + \frac{\gamma(\sum_{j=1}^{i} \lambda_j(1-\beta_j(\omega))) - \gamma(\sum_{j=1}^{i-1} \lambda_j(1-\beta_j(\omega)) + \omega)}{\omega - \lambda_i(1-\beta_i(\omega))} \right].$$

Remark 5.4.1
For $n = 1$, using (5.9), (5.10) and (5.13), (5.23) reduces to

$$\mathrm{E}(\mathrm{e}^{-\omega \mathbf{W}}) = \frac{(1-\rho)\omega}{\omega - \lambda(1-\beta(\omega))} \left[\frac{\mathrm{EI}}{s + \mathrm{EI}} + \frac{s}{s+\mathrm{EI}} \frac{1-\sigma(\omega)}{s\omega} \frac{\gamma(\omega)}{\sigma(\omega)} \right],$$

in which we recognize the well-known waiting-time decomposition property of $M/G/1$ vacation models.

□

From (5.23), using (5.9) and (5.12),

$$\mathrm{E}\mathbf{W}_i = \left[1 + 2\sum_{j=1}^{i-1} \rho_j + \rho_i \right] \frac{\mathrm{E}\bar{\mathbf{C}}^2}{2\mathrm{EC}} + \sum_{j=1}^{i-1} s_j = \tag{5.24}$$

$$\left[1 + 2\sum_{j=1}^{i-1} \rho_j + \rho_i \right] \frac{1}{1+\rho} \left[\frac{\sum_{j=1}^{n} \lambda_j \beta_j^{(2)}}{2(1-\rho)} + \frac{\rho s}{1-\rho} + \frac{s^{(2)}}{2(s + \frac{\gamma(\lambda)}{\lambda})} \right] + \sum_{j=1}^{i-1} s_j.$$

Remark 5.4.2
For $n = 1$ (5.24) reduces to

$$\mathrm{E}\mathbf{W} = [1+\rho]\frac{\mathrm{E}\bar{\mathbf{C}}^2}{2\mathrm{EC}} = \frac{\lambda\beta^{(2)}}{2(1-\rho)} + \frac{\rho s}{1-\rho} + \frac{s^{(2)}}{2(s + \frac{\gamma(\lambda)}{\lambda})},$$

which agrees with formula (5.40b) in Takagi [174] p. 213.

□

Remark 5.4.3

Noting that $\sum_{i=1}^{n} \rho_i[1 + 2\sum_{j=1}^{i-1} \rho_j + \rho_i] = \rho(1+\rho)$, we may obtain from (5.24) the pseudo-conservation law for the model under consideration,

$$\sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i = \rho(1+\rho)\frac{\mathrm{E}\bar{\mathbf{C}}^2}{2\mathrm{E}\mathbf{C}} + \sum_{i=1}^{n} \rho_i \sum_{j=1}^{i-1} s_j =$$

$$\rho\frac{\sum_{j=1}^{n} \lambda_j \beta_j^{(2)}}{2(1-\rho)} + \frac{\rho^2 s}{1-\rho} + \rho\frac{s^{(2)}}{2(s + \frac{\gamma(\lambda)}{\lambda})} + \sum_{i=1}^{n} \rho_i \sum_{j=1}^{i-1} s_j.$$

We may also obtain the pseudo-conservation law, without knowledge of the individual mean waiting times, using (5.9) and (5.10), from (4.15) with $GG = \{1, \ldots, n\}$, $\pi_1' = \dfrac{\mathrm{EI}}{\mathrm{EC}} = \dfrac{\gamma(\lambda)}{\lambda}\dfrac{1-\rho}{s + \frac{\gamma(\lambda)}{\lambda}}$, $\pi_1'' = 0$, $\pi_i = 0$, $i \neq 1$.

□

Remark 5.4.4

As seen from (5.24), the ordering of the queues involves an ordering of the mean waiting times,

$$\mathrm{E}\mathbf{W}_i - \mathrm{E}\mathbf{W}_{i+1} = -[\rho_i + \rho_{i+1}]\frac{\mathrm{E}\bar{\mathbf{C}}^2}{2\mathrm{E}\mathbf{C}} - s_i \leq 0, \qquad i = 1, \ldots, n-1.$$

The ordering of the queues even involves a stochastic ordering of the waiting times, as seen from the derivation of (5.23).

On the one hand the ordering of the waiting times may be argued to be unfair. Elevator polling (or scan polling) to some extent meets this objection to the globally gated service discipline. In a globally gated system with elevator polling the server alternatingly passes through 'up' cycles, visiting the queues in the order $Q_1, \ldots, Q_n$, and 'down' cycles, visiting the queues in the order $Q_n, \ldots, Q_1$, cf. Section 1.3. Thus as an immediate advantage elevator polling saves the switch-over time from Q_n to Q_1. At the start of each cycle, up or down, a similar centralized gating procedure is executed as described before. So alternatingly Q_1 and Q_n function as home base. Assuming the switch-over time from Q_{i+1} to Q_i to have the same mean as the switch-over time from Q_i to Q_{i+1}, Altman, Khamisy, & Yechiali [7] show that in a globally gated system with elevator polling the mean waiting times at all queues are equal, irrespective of the traffic characteristics! That fact once being known, it is less

surprising that a similar observation holds in the dormant server case. Obviously the waiting-time *distributions* at all queues are not equal. E.g., the variance of the waiting times is likely to be larger at the queues visited in the beginning or the end of a cycle than at the queues visited in the middle of a cycle.
On the other hand the ordering of the waiting times may be exploited to effectuate some kind of priorization. Following the line of this idea, similarly to Boxma, Levy, & Yechiali [50] simple index rules may be established for both static and dynamic optimization of the system performance, measured in terms of the mean waiting times.

□

As a justification of the dormant server policy, we now show the waiting time at each of the queues to be smaller (in the increasing-convex-ordering sense) than in the ordinary non-dormant server case. Let us label the stochastic variables and the associated LST's and pgf's in the dormant and non-dormant server case with a hat and a tilde, respectively. From [50] we have

$$\mathrm{E}\tilde{\mathbf{W}}_i = \left[1 + 2\sum_{j=1}^{i-1}\rho_j + \rho_i\right]\frac{\mathrm{E}\tilde{\mathbf{C}}^2}{2\mathrm{E}\tilde{\mathbf{C}}} + \sum_{j=1}^{i-1}s_j = \tag{5.25}$$

$$\left[1 + 2\sum_{j=1}^{i-1}\rho_j + \rho_i\right]\frac{1}{1+\rho}\left[\frac{\sum_{j=1}^{n}\lambda_j\beta_j^{(2)}}{2(1-\rho)} + \frac{\rho s}{1-\rho} + \frac{s^{(2)}}{2s}\right] + \sum_{j=1}^{i-1}s_j.$$

Subtracting (5.25) from (5.24),

$$\mathrm{E}\hat{\mathbf{W}}_i - \mathrm{E}\tilde{\mathbf{W}}_i = -\left[1 + 2\sum_{j=1}^{i-1}\rho_j + \rho_i\right]\frac{1}{1+\rho}\frac{\frac{\hat{\gamma}(\lambda)}{\lambda}}{s + \frac{\hat{\gamma}(\lambda)}{\lambda}}\frac{s^{(2)}}{2s} \leq 0.$$

Proceeding by differentiating the LST of the waiting-time distribution not just once but several times, we may prove that in fact not only the mean waiting times are smaller, but also each of the higher-order moments, i.e., $\mathrm{E}(\hat{\mathbf{W}}_i^k) \leq \mathrm{E}(\tilde{\mathbf{W}}_i^k)$ for any $k \geq 1$. By using coupling techniques we may however prove that the waiting times are in fact even smaller in the increasing-convex-ordering sense, i.e., $\mathrm{E}f(\hat{\mathbf{W}}_i) \leq \mathrm{E}f(\tilde{\mathbf{W}}_i)$ for any non-decreasing convex function $f(\cdot)$.

Lemma 5.4.1
$\hat{\mathbf{W}}_i \leq_{\mathrm{icx}} \tilde{\mathbf{W}}_i, \qquad i = 1, \ldots, n,$

i.e., $\mathrm{E}f(\hat{\mathbf{W}}_i) \leq \mathrm{E}f(\tilde{\mathbf{W}}_i)$, $i = 1, \ldots, n$, for any non-decreasing convex function $f(\cdot)$.

Proof
See Appendix 5.B.

□

The ordering relation stated in the above lemma adds to the modest collection of stochastic ordering results that are known for polling systems so far. The scarce results that are known exclusively refer to stochastic monotonicity properties of *global* performance measures, like the total amount of work in the system or the cycle time, or refer to monotonicity of quantities like the joint queue length at polling epochs with regard to the parameters of the service discipline, or with regard to the underlying stochastic processes. Levy et al. [142] showed that the total amount of work in the system is decreasing in the degree of exhaustiveness of the service discipline. Altman et al. [8] proved that the cycle time and the joint queue length at polling epochs are stochastically increasing in the arrival rates, service times, and switch-over times. To the best of the author's knowledge, there are however no ordering results known at all for the individual waiting times of the nature of the ordering relation stated in the above lemma. One might be inclined to conjecture that also the individual waiting times are stochastically decreasing in the degree of exhaustiveness of the service discipline or increasing in the arrival rates, service times, and switch-over times, but such statements have either been disproved by simple counterexamples or have lacked proof so far.

5.5 The queue length

In Section 5.3 we expressed the pgf of the joint queue length distribution at the beginning of a cycle into the LST of the restricted cycle time distribution. In this section we relate the joint queue length distribution at polling epochs to the joint queue length distribution at the beginning of a cycle. Thus we obtain an expression for the pgf of the joint queue length distribution at polling epochs in terms of the LST of the restricted cycle time distribution. We analyze both the queue lengths at $Q_1, \ldots, Q_n$ at a polling epoch at Q_i, and the queue lengths at $Q_1, \ldots, Q_n$ seen by the server at successive polling epochs at $Q_1, \ldots, Q_n$. As described in Section 2.2, the marginal queue length distribution at Q_i at an arbitrary epoch may be derived from the waiting-time distribution at Q_i as obtained in (5.23).

We first introduce some notation. Denote by $\mathbf{X}_{ij}$ the number of customers at Q_j at a polling epoch at Q_i, i.e., at the start of a visit to Q_i, $i = 1, \ldots, n$, $j = 1, \ldots, n$. Denote by $\mathbf{D}_i$ the indicator function of the event that an arbitrary customer (that arrives in an empty system) arrives at Q_i, i.e., a stochastic variable which is 1 with probability λ_i/λ and 0 with probability $1 - \lambda_i/\lambda$, $i = 1, \ldots, n$.

By the nature of the globally gated service discipline $\mathbf{X}_{ij}$ and $\mathbf{Y}_j$, $i = 1, \ldots, n$, $j = 1, \ldots, n$, are related as follows.

$$\mathbf{X}_{ij} \stackrel{d}{=} \mathbf{Y}_j(i \le j) + \mathbf{A}_j(\sum_{h=1}^{i-1}(\mathbf{V}(\mathbf{Y}_h) + \mathbf{S}_h)) + \tag{5.26}$$

$$\left(\mathbf{D}_j(i \le j) + \mathbf{A}_j(\sum_{h=1}^{i-1}\mathbf{D}_h\bar{\mathbf{C}}_h)\right)\mathrm{I}_{\{\mathbf{Y}=0\}},$$

with $\mathrm{I}_{\{\mathbf{Y}=0\}}$ denoting the indicator function of the event $(\mathbf{Y}_1,\ldots,\mathbf{Y}_n) = (0,\ldots,0)$ and $(i \le j)$ is equal to 1 if $i \le j$ and 0 if $i > j$. The notation was further introduced in Section 5.4.
We first study the joint distribution of $\mathbf{X}_{i1},\ldots,\mathbf{X}_{in}$, the queue lengths at $Q_1,\ldots,Q_n$ at a polling epoch at Q_i, $i = 1,\ldots,n$.
From (5.26),

$$\mathrm{E}(z_1^{\mathbf{X}_{i1}}\ldots z_n^{\mathbf{X}_{in}} \mid \mathbf{Y}_1,\ldots,\mathbf{Y}_n) = \tag{5.27}$$

$$\prod_{h=1}^{i-1}\sigma_h(\epsilon(z)) \times \left[\prod_{h=1}^{i-1}\beta_h(\epsilon(z))^{\mathbf{Y}_h}\prod_{h=i}^{n}z_h^{\mathbf{Y}_h} - \right.$$

$$\left.\left(1 - \sum_{h=1}^{i-1}\frac{\lambda_h}{\lambda}\beta_h(\epsilon(z)) - \sum_{h=i}^{n}\frac{\lambda_h}{\lambda}z_h\right)\mathrm{I}_{\{\mathbf{Y}=0\}}\right],$$

with $\epsilon(z_1,\ldots,z_n) = \sum_{j=1}^{n}\lambda_j(1-z_j)$, $\mid z_j \mid \le 1$, $j = 1,\ldots,n$.
Unconditioning (5.27), using (5.6),

$$\mathrm{E}(z_1^{\mathbf{X}_{i1}}\ldots z_n^{\mathbf{X}_{in}}) = \tag{5.28}$$

$$\prod_{h=1}^{i-1}\sigma_h(\epsilon(z)) \times \left[\gamma(\sum_{h=1}^{i-1}\lambda_h(1-\beta_h(\epsilon(z)) + \sum_{h=i}^{n}\lambda_h(1-z_h)) - \right.$$

$$\left.\frac{\gamma(\lambda)}{\lambda}\left(\sum_{h=1}^{i-1}\lambda_h(1-\beta_h(\epsilon(z))) + \sum_{h=i}^{n}\lambda_h(1-z_h)\right)\right].$$

For $i = n+1$, using (5.6) and interpreting $\mathbf{X}_{n+1,j}$ as $\mathbf{Y}_j$, $j = 1,\ldots,n$, (5.28) reduces to (5.8).
From (5.28),

$$\mathrm{Cov}(\mathbf{X}_{ik},\mathbf{X}_{il}) = \lambda_k\lambda_l \times \left[\mathrm{Var}(\sum_{h=1}^{i-1}\mathbf{S}_h) + \mathrm{E}\mathbf{C}\sum_{h=1}^{i-1}\lambda_h\beta_h^{(2)} + \right. \tag{5.29}$$

$$\left.(\mathrm{E}\bar{\mathbf{C}}^2 - (\mathrm{E}\mathbf{C})^2)(\sum_{h=1}^{i-1}\rho_h + (i \le k))(\sum_{h=1}^{i-1}\rho_h + (i \le l))\right].$$

For $i = n+1$, using (5.12) and interpreting $\mathbf{X}_{n+1,j}$ as $\mathbf{Y}_j$, $j = 1, \ldots, n$, (5.29) reduces to

$$\mathrm{Cov}(\mathbf{Y}_k, \mathbf{Y}_l) = \lambda_k \lambda_l (\mathrm{E}\bar{\mathbf{C}}^2 - (\mathrm{E}\bar{\mathbf{C}})^2),$$

which may also be obtained from (5.6).

We finally study the joint distribution of $\mathbf{X}_{11}, \ldots, \mathbf{X}_{nn}$, the queue lengths at $Q_1, \ldots, Q_n$ seen by the server at successive polling epochs at $Q_1, \ldots, Q_n$. From (5.26),

$$\mathrm{E}(z_1^{\mathbf{X}_{11}} \ldots z_n^{\mathbf{X}_{nn}} \mid \mathbf{Y}_1, \ldots, \mathbf{Y}_n) = \tag{5.30}$$

$$\prod_{i=1}^{n} \sigma_i\Big(\sum_{h=i+1}^{n} \lambda_h(1-z_h)\Big) \times \Bigg[\prod_{i=1}^{n} \beta_i\Big(\sum_{h=i+1}^{n} \lambda_h(1-z_h)\Big)^{\mathbf{Y}_i} \prod_{i=1}^{n} z_i^{\mathbf{Y}_i} - \left(1 - \sum_{i=1}^{n} \frac{\lambda_i}{\lambda} z_i \beta_i\Big(\sum_{h=i+1}^{n} \lambda_h(1-z_h)\Big)\right) \mathrm{I}_{\{\mathbf{Y}=0\}}\Bigg].$$

Unconditioning (5.30), using (5.6),

$$\mathrm{E}(z_1^{\mathbf{X}_{11}} \ldots z_n^{\mathbf{X}_{nn}}) = \tag{5.31}$$

$$\prod_{i=1}^{n} \sigma_i\Big(\sum_{h=i+1}^{n} \lambda_h(1-z_h)\Big) \times \Bigg[\gamma\Big(\sum_{i=1}^{n} \lambda_i\Big(1 - z_i\beta_i\Big(\sum_{h=i+1}^{n} \lambda_h(1-z_h)\Big)\Big)\Big) - \frac{\gamma(\lambda)}{\lambda}\left(\sum_{i=1}^{n} \lambda_i\Big(1 - z_i\beta_i\Big(\sum_{h=i+1}^{n} \lambda_h(1-z_h)\Big)\Big)\right)\Bigg].$$

From (5.31),

$$\mathrm{Cov}(\mathbf{X}_{kk}, \mathbf{X}_{ll}) = \lambda_k \lambda_l \times \Bigg[\mathrm{Cov}\Big(\sum_{i=1}^{k-1} \mathbf{S}_i, \sum_{i=1}^{l-1} \mathbf{S}_i\Big) + \tag{5.32}$$

$$\mathbf{EC}\left(q(k,l) + \sum_{i=1}^{\min(k,l)-1} \lambda_i \beta_i^{(2)}\right) + (\mathrm{E}\bar{\mathbf{C}}^2 - (\mathbf{EC})^2)\Big(\sum_{i=1}^{k-1} \rho_i + 1\Big)\Big(\sum_{i=1}^{l-1} \rho_i + 1\Big)\Bigg],$$

where $q(k,l)$ is equal to β_k if $k < l$, β_l if $k > l$ and 0 if $k = l$.

Remark 5.5.1
Unlike in the ordinary non-dormant server case, cf. Boxma, Weststrate, & Yechiali [41], neither the queue lengths at a polling epoch, nor the queue lengths seen by the server at successive polling epochs need to be positively correlated. E.g., if all service times and switch-over times are zero, then (5.29) and (5.32) reduce to

$$\mathrm{Cov}(\mathbf{X}_{ik}, \mathbf{X}_{il}) = -\frac{\lambda_k \lambda_l}{\lambda^2}(i \leq k)(i \leq l),$$

and

$$\mathrm{Cov}(\mathbf{X}_{kk}, \mathbf{X}_{ll}) = -\frac{\lambda_k \lambda_l}{\lambda^2},$$

respectively. This may also be seen immediately, as during every cycle exactly one customer is served, which occurs at Q_i with probability λ_i/λ, $i = 1, \ldots, n$. □

Appendices

5.A Proof of Lemma 5.3.1

Lemma 5.3.1

i. $\lim_{K \to \infty} \delta^{(K)}(\omega) = 0$ for all ω with $\mathrm{Re}\,\omega \geq 0$.

ii. $\prod_{k=0}^{\infty} \sigma(\delta^{(k)}(\omega))$ converges for all ω with $\mathrm{Re}\,\omega \geq 0$.

iii. $\sum_{k=0}^{\infty} \delta^{(k+1)}(\omega) \prod_{l=0}^{k} \sigma(\delta^{(l)}(\omega))$ converges for all ω with $\mathrm{Re}\,\omega \geq 0$.

Proof

For any non-negative valued stochastic variable $\mathbf{H}$, having distribution $H(\cdot)$ with LST $\eta(\cdot)$, holds

$$\begin{aligned} |\, 1 - \eta(\omega) \,| &= \Big|\, \int_{t=0}^{\infty} (1 - \mathrm{e}^{-\omega t}) dH(t) \,\Big| \\ &\leq \int_{t=0}^{\infty} |\, 1 - \mathrm{e}^{-\omega t} \,|\, dH(t) \\ &\leq \int_{t=0}^{\infty} |\, \omega t \,|\, dH(t) \\ &= \mathrm{E}\mathbf{H}\, |\, \omega \,|, \end{aligned} \tag{5.33}$$

for all ω with $\mathrm{Re}\,\omega \geq 0$, since $|\, 1 - \mathrm{e}^{-z} \,| \leq |\, z \,|$ for all z with $\mathrm{Re}\, z \geq 0$.

Proof of i.

$$\mathrm{Re}\,\delta^{(k)}(\omega) \geq 0, \qquad k = 0, 1, \ldots, \tag{5.34}$$

for all ω with $\mathrm{Re}\,\omega \geq 0$.

Further, using (5.33),

$$|\, \delta(\omega) \,| = \Big|\, \sum_{i=1}^{n} \lambda_i (1 - \beta_i(\omega)) \,\Big|$$

$$\begin{aligned} &\leq \sum_{i=1}^{n} \lambda_i \mid 1 - \beta_i(\omega) \mid \\ &\leq \sum_{i=1}^{n} \lambda_i \beta_i \mid \omega \mid \\ &= \rho \mid \omega \mid, \end{aligned}$$

for all ω with $\mathrm{Re}\,\omega \geq 0$.
Hence, by induction, using (5.34),

$$\mid \delta^{(k)}(\omega) \mid \leq \rho^k \mid \omega \mid, \qquad k = 0, 1, \ldots, \tag{5.35}$$

for all ω with $\mathrm{Re}\,\omega \geq 0$.
As $\rho < 1$ is assumed to hold, we conclude that $\lim_{K \to \infty} \delta^{(K)}(\omega) = 0$ for all ω with $\mathrm{Re}\,\omega \geq 0$.
Proof of ii.
From the theory of infinite products, cf. Titchmarsh [185] p. 18, we have that $\prod_{k=0}^{\infty} \sigma(\delta^{(k)}(\omega))$ converges iff $\sum_{k=0}^{\infty} [1 - \sigma(\delta^{(k)}(\omega))]$ converges.
Using (5.33), (5.35), (5.34), and the assumption that $\rho < 1$,

$$\begin{aligned} \mid \sum_{k=0}^{\infty} [1 - \sigma(\delta^{(k)}(\omega))] \mid &\leq \sum_{k=0}^{\infty} \mid 1 - \sigma(\delta^{(k)}(\omega)) \mid \\ &\leq \sum_{k=0}^{\infty} s \mid \delta^{(k)}(\omega) \mid \\ &\leq s \sum_{k=0}^{\infty} \rho^k \mid \omega \mid \\ &= \frac{s}{1-\rho} \mid \omega \mid \\ &< \infty, \end{aligned}$$

for all ω with $\mathrm{Re}\,\omega \geq 0$. So $\prod_{k=0}^{\infty} \sigma(\delta^{(k)}(\omega))$ converges for all ω with $\mathrm{Re}\,\omega \geq 0$.
Proof of iii.
Because of (5.34),

$$\mid \sigma(\delta^{(k)}(\omega)) \mid \leq 1, \qquad k = 0, 1, \ldots, \tag{5.36}$$

for all ω with $\mathrm{Re}\,\omega \geq 0$. Using (5.35), (5.36), and the assumption that $\rho < 1$,

$$\begin{aligned} \mid \sum_{k=0}^{\infty} \delta^{(k+1)}(\omega) \prod_{l=0}^{k} \sigma(\delta^{(l)}(\omega)) \mid &\leq \sum_{k=0}^{\infty} \mid \delta^{(k+1)}(\omega) \mid \prod_{l=0}^{k} \mid \sigma(\delta^{(l)}(\omega)) \mid \\ &\leq \sum_{k=0}^{\infty} \rho^{k+1} \mid \omega \mid \end{aligned}$$

$$\begin{aligned} &= \frac{\rho}{1-\rho} \mid \omega \mid \\ &< \infty, \end{aligned}$$

for all ω with $\mathrm{Re}\,\omega \geq 0$. So $\sum_{k=0}^{\infty} \delta^{(k+1)}(\omega) \prod_{l=0}^{k} \sigma(\delta^{(l)}(\omega))$ converges for all ω with $\mathrm{Re}\,\omega \geq 0$.

□

5.B Proof of Lemma 5.4.1

Lemma 5.4.1
$\hat{\mathbf{W}}_i \leq_{\mathrm{icx}} \tilde{\mathbf{W}}_i, \qquad i = 1, \ldots, n,$

i.e., $\mathrm{E}f(\hat{\mathbf{W}}_i) \leq \mathrm{E}f(\tilde{\mathbf{W}}_i)$, $i = 1, \ldots, n$, for any non-decreasing convex function $f(\cdot)$.

Proof
We sketch the intuitive idea of the proof. For a detailed technical proof we refer to appendix B of Borst [26]. We assume the arrival, service, and switch-over processes in the dormant and non-dormant server case to be coupled as follows. In both cases the server experiences exactly the same switch-over times, but - since the dormant and non-dormant server case evolve according to different operational rules - the same switch-over time is not necessarily experienced at the same point in time, i.e., the switch-over times may be shifted in time. Moreover, in the dormant server case, when the server is actually idling, we assume that the server is experiencing a switch-over time, which is however immediately interrupted as soon as a new customer arrives, just as if the server would have been idling, awaiting a new customer to arrive. The remainder of the switch-over time is then resumed as soon as the server starts idling again. During one and the same switch-over time the arrival processes in both cases proceed synchronously, i.e., the same customer arrives at the same relative time (with regard to the switch-over time in question), requiring the same service time. Thus the server also provides exactly the same service times in both cases, but - since the dormant and non-dormant server case evolve according to different operational rules - the same service time is not necessarily provided at the same point in time. Also during one and the same service time the arrival processes in both cases proceed synchronously. So the arrivals in both cases may be shifted in time, however, congruently to the service or switch-over times in which they fall, so that the same customer arrives at the same relative time with regard to the service time or switch-over time in question. By the memoryless property of the Poisson process the coupling does not affect the stochastic properties of the arrival process. Neither does the coupling affect the stochastic properties of the service and switch-over processes. Thus we obtain coupled but still marginally unbiased induced stochastic processes (like waiting times and queue lengths) in the dormant and non-dormant server case.

Suppose that at time $t = \mathbf{T}_0$ in both cases the system is empty and the server is at its home base Q_1, just back from switching. The server then starts switching for a time of length $\mathbf{S}_0$. $\mathbf{S}_0$, $\mathbf{S}_1$, $\mathbf{S}_2$, ... are independent stochastic variables with common distribution $S(\cdot)$. During the switch-over time $\mathbf{S}_0$ a number of $\mathbf{K}$ customers arrive, let us say $C_1, \ldots, C_{\mathbf{K}}$, at (relative) time $t = \mathbf{A}_1$, $t = \mathbf{A}_1 + \mathbf{A}_2$, ..., $t = \mathbf{A}_1 + \ldots + \mathbf{A}_{\mathbf{K}}$, requiring service times of length $\mathbf{B}_1$, $\mathbf{B}_2$, ..., $\mathbf{B}_{\mathbf{K}}$. $\mathbf{A}_1$, $\mathbf{A}_2$, ... are independent exponentially distributed stochastic variables with mean $1/\lambda$. $\mathbf{B}_1$, $\mathbf{B}_2$, ... are independent stochastic variables with common distribution $\sum_{i=1}^{n} \frac{\lambda_i}{\lambda} B_i(\cdot)$.

In the dormant server case, at time $t = \mathbf{T}_0 + \mathbf{A}_1$ the server interrupts switching, suspending the remainder $\mathbf{S}_0 - \mathbf{A}_1$ of the switch-over time $\mathbf{S}_0$, and starts a cycle along the queues to serve the newly arrived customer C_1, just as if the server would have remained idling at Q_1 from time $t = \mathbf{T}_0$ on, awaiting the new customer to arrive. At time $t = \mathbf{T}_1$, after $\mathbf{L}_1$ cycles, the system is empty again and the server is back again at its home base Q_1 (these events occurring simultaneously for the first time). The server then starts switching, resuming the switch-over time $\mathbf{S}_0$.

At time $t = \mathbf{T}_1 + \mathbf{A}_2$ the server again interrupts switching, suspending the remainder $\mathbf{S}_0 - \mathbf{A}_1 - \mathbf{A}_2$ of the switch-over time $\mathbf{S}_0$, and starts a cycle along the queues to serve the newly arrived customer C_2, again just as if the server would have remained idling at Q_1 from time $t = \mathbf{T}_1$ on, awaiting the new customer to arrive.

In the non-dormant server case, at time $t = \mathbf{T}_0 + \mathbf{A}_1$ the server just continues switching, disregarding the newly arrived customer C_1. At time $t = \mathbf{T}_0 + \mathbf{S}_0$ the server finishes switching and starts a cycle along the queues to serve the newly arrived customers $C_1, \ldots, C_{\mathbf{K}}$.

In the dormant server case, at time $t = \mathbf{T}_{\mathbf{K}}$, after $\mathbf{L}_1 + \ldots + \mathbf{L}_{\mathbf{K}}$ cycles, the system is empty and the server is back at its home base Q_1 (these events occurring simultaneously for the $\mathbf{K}$-th time). The server then starts switching, resuming the switch-over time $\mathbf{S}_0$. At time $t = \mathbf{U}_0$ the server finishes switching, $\mathbf{U}_0 = \mathbf{T}_{\mathbf{K}} + \mathbf{D}$, $\mathbf{D} = \mathbf{S}_0 - \mathbf{A}_1 - \ldots - \mathbf{A}_{\mathbf{K}}$. At time $t = \mathbf{U}_0$ also in the non-dormant server case the system is empty and the server just finishes switching. In both cases the server has then experienced exactly the same switch-over times, viz. $\mathbf{S}_0$, $\mathbf{S}_1$, ..., $\mathbf{S}_{\mathbf{L}_1+\ldots+\mathbf{L}_{\mathbf{K}}}$, and has provided exactly the same service times, viz. the service times of the customers arriving during $\mathbf{S}_0$, $\mathbf{S}_1$, ..., $\mathbf{S}_{\mathbf{L}_1+\ldots+\mathbf{L}_{\mathbf{K}}}$, and of their descendants. Let us say the total number of type-i customers among them is $\mathbf{M}_i$. (Here the descendants of a customer are recursively defined as the customers arriving during its service time or during the service time of one of its descendants.) Concluding, at time $t = \mathbf{U}_0$ in *both* cases the system is empty and the server is at its home base Q_1, just back from switching.

Let $R_i^{(h)}$ be the h-th type-i customer served from time $t = \mathbf{T}_0$ on in the dormant server case, $h = 1, 2, \ldots$. Denote by $\hat{\mathbf{W}}_i^{(h)}$ and $\tilde{\mathbf{W}}_i^{(h)}$ the waiting time

of $R_i^{(h)}$ in the dormant and non-dormant server case respectively, $h = 1, 2, \ldots$. As the stochastic processes $\{\hat{\mathbf{W}}_i^{(h)}, h = 1, 2, \ldots\}$ and $\{\tilde{\mathbf{W}}_i^{(h)}, h = 1, 2, \ldots\}$ are regenerative with regard to $h = 1$ and $h = \mathbf{M}_i + 1$,

$$\mathrm{E}f(\hat{\mathbf{W}}_i) = \frac{1}{\mathrm{E}\mathbf{M}_i}\mathrm{E}(\sum_{h=1}^{\mathbf{M}_i} f(\hat{\mathbf{W}}_i^{(h)})), \tag{5.37}$$

and

$$\mathrm{E}f(\tilde{\mathbf{W}}_i) = \frac{1}{\mathrm{E}\mathbf{M}_i}\mathrm{E}(\sum_{h=1}^{\mathbf{M}_i} f(\tilde{\mathbf{W}}_i^{(h)})). \tag{5.38}$$

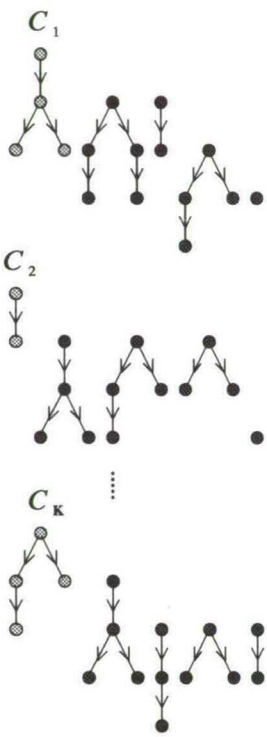

FIGURE 5.1. A dormant server.

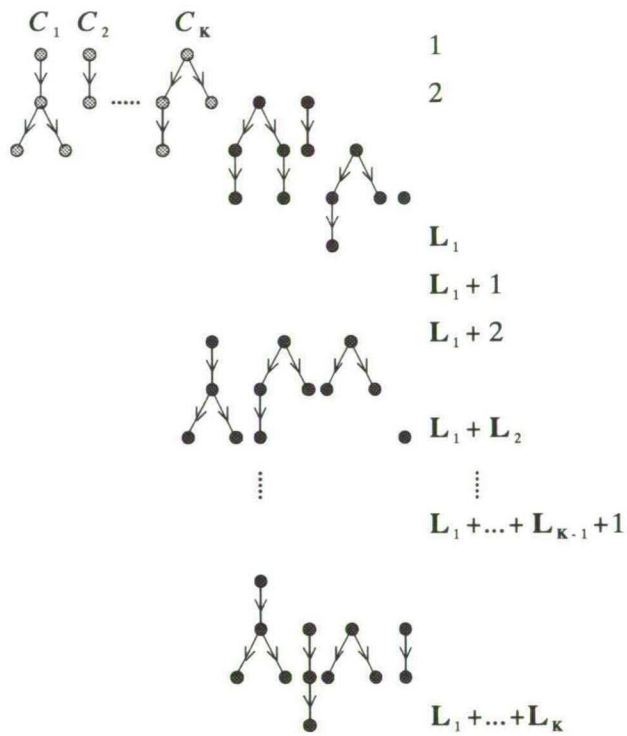

FIGURE 5.2. A non-dormant server.

Consider now Figure 5.1 and Figure 5.2, representing the customer offspring process in the dormant and non-dormant server case, respectively. In both cases, dots at the same (horizontal) level correspond to customers served in the same cycle. An arc indicates that the customer at the head arrives during the service time of the customer at the tail. A dot without any incoming arc represents a customer which arrives during a switch-over time, or, in the dormant

server case, during an idling period. A dot without any outgoing arc represents a customer requiring a service time during which no single customer arrives.
To prove that (5.38) majorizes (5.37), we need to introduce some additional terminology. In the dormant server case the interval from time $t = \mathbf{T}_{k-1}$ to time $t = \mathbf{T}_k$, comprising the cycles $\mathbf{L}_1 + \ldots + \mathbf{L}_{k-1} + 1$ to $\mathbf{L}_1 + \ldots + \mathbf{L}_k$, is referred to as the k-th *busy interval*, $k = 1, \ldots, \mathbf{K}$. Customers arriving during $\mathbf{S}_0$ (thus interrupting $\mathbf{S}_0$ in the dormant server case), together with their descendants, are called *primary* customers (corresponding to the light-grey dots in Figure 5.1 and Figure 5.2. The remaining customers, i.e., customers arriving during $\mathbf{S}_1, \ldots, \mathbf{S}_{\mathbf{L}_\mathbf{K}}$, together with their descendants, are called *secondary* customers (the dark-grey dots).
With the busy intervals as background, the dormant and non-dormant server case differ in the service of primary customers, but not in the service of secondary customers. In the dormant server case the service of primary customers is balanced over the $\mathbf{K}$ busy intervals. In the non-dormant server case the service of primary customers is concentrated in the first busy interval. The service of secondary customers occurs in both cases in corresponding cycles. So in the non-dormant server case the primary customers all bother one another and all bother the same secondary customers. To make the latter intuitive idea precise, observe that the waiting time of every customer is composed of switch-over time, service time of primary customers, and service time of secondary customers. For any waiting time $\mathbf{W}_i^{(h)}$, denote by $\mathbf{V}_{i,k}^{(h)}$ the share constituted by service time of primary customers that are descendants of C_k, $k = 1, \ldots, \mathbf{K}$. Denote by $\mathbf{V}_i^{(h)}$ the remaining share in $\mathbf{W}_i^{(h)}$, i.e., the share constituted by switch-over time and service time of secondary customers.
For a *primary* customer $R_i^{(h)}$ we have

$$\hat{\mathbf{V}}_i^{(h)} + \mathbf{Z}_i^{(h)} \stackrel{d}{=} \tilde{\mathbf{V}}_i^{(h)}, \qquad \mathbf{Z}_i^{(h)} \geq 0. \tag{5.39}$$

Assuming that $R_i^{(h)}$ is a descendant of C_k,

$$\hat{\mathbf{V}}_{i,k}^{(h)} = \tilde{\mathbf{V}}_{i,k}^{(h)}, \qquad \hat{\mathbf{V}}_{i,m}^{(h)} = 0, \qquad m \neq k. \tag{5.40}$$

For a *secondary* customer $R_i^{(h)}$ we have

$$\hat{\mathbf{V}}_i^{(h)} = \tilde{\mathbf{V}}_i^{(h)}. \tag{5.41}$$

Let $H_{i,l}$ be the index set of the secondary type-i customers served in the l-th cycle in the dormant server case, $l = 1, \ldots, \mathbf{L}_1 + \ldots + \mathbf{L}_\mathbf{K}$. Let $H_{i,kl}$ be the index set of the secondary type-i customers served in the $\mathbf{L}_{k-1} + l$-th cycle in the dormant server case if $l \leq \mathbf{L}_k$, $k = 1, \ldots, \mathbf{K}$. If $l > \mathbf{L}_k$, let $H_{i,kl}$ be the empty set. We then have

$$\sum_{h \in H_{i,kl}} \hat{\mathbf{V}}_{i,k}^{(h)} \stackrel{d}{=} \sum_{h \in H_{i,l}} \tilde{\mathbf{V}}_{i,k}^{(h)}, \qquad \sum_{h \in H_{i,kl}} \hat{\mathbf{V}}_{i,m}^{(h)} = 0, \qquad m \neq k, \tag{5.42}$$

The relations (5.39), (5.40), (5.41), and (5.42) constitute the key elements in proving that (5.38) majorizes (5.37). For a detailed comparison we refer to appendix B of Borst [26].

□

Chapter 6

Optimization of k-limited service strategies

6.1 Introduction

In the present chapter we consider a polling system with a k-limited service strategy. Under k-limited service, when visiting a queue, the server works until either a prespecified number of k customers have been served, or the queue becomes empty, whichever occurs first. We are interested in the problem of determining appropriate values for the service limits that contribute to an efficient operation of the system.

As described in Section 1.2, polling systems may be used to model communication systems with a conflict-free medium access protocol. The standard control mechanism to regulate the operation of those systems is either to limit the *number* of transmissions or to limit the transmission *time* granted to the stations. As a major benefit, by setting such service limits one may accomplish a bound on the cycle time, i.e., the time until a station acquires the right of transmission again. Moreover, one may effectuate some kind of priorization by assigning different service limits to different stations, according to their relative importance.

Although the use of service limits is very standard, it is still not very well understood how the limits should be set so as to optimize the performance of such communication systems. Driven by these considerations we address in the present chapter the problem how in a polling system with k_i-limited service at Q_i the limits should be set so as to minimize the waiting cost as performance measure. It appears that if we do not impose any constraint on the k_i's, then at least one of the optimal k_i's is always infinite. To accomplish a bound on the cycle time we therefore also study a version of the problem with a constraint

of the form $\sum_{i=1}^{n} \gamma_i k_i \leq K$.

Although similar in their operation, limits on the visit time are analytically even harder to handle than limits on the number of services during a visit. Therefore, k_i-limited service is frequently used as an approximation of T_i-limited service, with $k_i \approx T_i/\beta_i$, the exact value depending on whether or not service is preempted when the timer expires. Consequently, the optimal k_i's are likely to provide a reasonable indication for the optimal T_i's. Note that for deterministic service times, corresponding to the practically relevant case of transmission of packets of fixed length, k_i-limited service and T_i-limited service even coincide. Blanc & Van der Mei [23] consider a similar optimization problem in a polling system with Bernoulli service. Under Bernoulli service with parameter q_i, when visiting Q_i, S serves at least one customer, and at every service completion S serves yet another customer with probability q_i, and leaves Q_i with probability $1 - q_i$; S also leaves Q_i when it becomes empty. The number of potential services being geometrically distributed with mean $1/(1-q_i)$, Bernoulli service may be used as an emulation of k_i-limited service, with $q_i = 1 - 1/k_i$. Note that Bernoulli service and k_i-limited service even coincide for $q_i = 0$ as well as $q_i = 1$.

Polling systems with limited-type service strategies are extremely hard to analyze, not to mention optimize. As noted in Section 1.4, limited-type service policies do not satisfy Property 1.4.1, which provides a global characterization of the class of service disciplines that are amenable to an exact analysis. Indeed, polling systems with limited-type service strategies have not allowed an exact analysis, apart from some special cases like two-queue cases and completely symmetric cases. Eisenberg [85] studies a two-queue model with zero switch-over times and 1-limited service at both queues, transforming the problem of finding the joint queue length distribution into the problem of solving a singular Fredholm integral equation. Cohen & Boxma [74] analyze the same model, translating the problem into a Riemann-Hilbert boundary value problem. Using similar techniques, Boxma [37] studies a symmetric two-queue model with non-zero switch-over times and 1-limited service at both queues. Boxma & Groenendijk [43] analyze an asymmetric two-queue model with non-zero switch-over times and 1-limited service at both queues by formulating a Riemann boundary value problem. Fuhrmann [100] derives the mean waiting time in a completely symmetric system with 1-limited service and an arbitrary number of queues. The two-queue model with 1-limited service at one queue and *exhaustive* service at the other has turned out to be relatively simple to analyze, cf. Groenendijk [114] Section 6.3. The two-queue model with 1-limited service at one queue and *gated* service at the other however appears to defy an exact analysis. Polling systems with time-limited service have not yielded to an exact analysis either. See Coffman et al. [71] for an interesting analysis of a two-queue model with exponentially distributed time limits.

The fact that limited-type service policies are extremely hard to analyze, has considerably added to the importance of pseudo-conservation laws, which pro-

vide relatively simple exact expressions for a specific weighted sum of the mean waiting times, cf. Section 2.3. Thus pseudo-conservation laws are an important instrument in constructing and validating waiting-time approximations. For the case of 1-limited service at each of the queues the pseudo-conservation law was originally derived in Watson [187]. As described in Section 2.3, Boxma & Groenendijk [42] developed an approach to obtain a pseudo-conservation law in a considerably more general framework, covering 1-limited service at each of the queues. Boxma & Meister [52] use the pseudo-conservation law to derive waiting-time approximations for 1-limited service. Groenendijk [114] presents a more refined procedure to compute such approximations. For the general case of k-limited service the pseudo-conservation law still contains an unknown term. Fuhrmann & Wang [104] obtain waiting-time approximations for k-limited service by bounding that term. Everitt [87], [89] derives such approximations by approximating that term. Chang & Sandhu [66] present a more refined procedure to calculate waiting-time approximations for k-limited service.

The remainder of the chapter is organized as follows. In Section 6.2 we present a detailed model description and formulate a pseudo-conservation law for the mean waiting times. In Section 6.3 we propose four different approximative approaches to the constrained optimization problem, based on four different approximations for the mean waiting times. These approaches are numerically investigated in Section 6.4. In Section 6.5 we discuss some properties of polling systems with k-limited service and establish a (partially conjectured) $c\mu$-like rule for the unconstrained optimization problem. We then also propose an approximative approach to this problem. This approach is numerically examined in Section 6.6. In Section 6.7 we conclude with some remarks and suggestions for further research.

6.2 Model description and preliminaries

The model under consideration consists of n queues, $Q_1, \ldots, Q_n$, each of infinite capacity, attended by a single server S. For the specification of the arrival, service, and switch-over processes we refer to the description of the 'basic model' in Section 1.3.

The server visits the queues in strictly cyclic order, $Q_1, \ldots, Q_n$. When visiting Q_i, S works until either k_i customers have been served or the queue becomes empty, whichever occurs first. Note that the case $k_i = \infty$ amounts to ordinary exhaustive service. Fuhrmann & Wang [104] call this discipline exhaustive-limited service, as opposed to gated-limited service in case of which S, when meeting l_i customers at Q_i upon arrival, only serves $\min\{l_i, k_i\}$ of them. In case of G-limited service, $k_i = \infty$ amounts to ordinary gated service.

As mentioned in Section 1.3, a necessary and sufficient condition for stability is $\lambda_i s/(1-\rho) < k_i$, $i = 1, \ldots, n$. Throughout the chapter the stability condition is assumed to hold.

Denote by $\mathbf{W}_i$ the waiting time of an arbitrary type-i customer, $i = 1, \ldots, n$. Let c_i represent the waiting cost per unit of time of a type-i customer, $i = 1, \ldots, n$. The mean total waiting cost per unit of time amounts to $\sum_{i=1}^{n} c_i \lambda_i \mathrm{E}\mathbf{W}_i$. In this chapter we are interested in the problem of finding the values for $k_1, \ldots, k_n$ that minimize that quantity, both for the constrained case with $\sum_{i=1}^{n} \gamma_i k_i \leq K$ and for the unconstrained case. Note that taking $\gamma_i \equiv 1$ puts a limit on the number of services in a cycle. Choosing $\gamma_i \equiv \beta_i$ yields, for deterministic service and switch-over times, a bound on the cycle time.
Everitt [87] has derived the following pseudo-conservation law for the mean waiting times:

$$\sum_{i=1}^{n} \rho_i \left(1 - \frac{\lambda_i s}{k_i(1-\rho)}\right) \mathrm{E}\mathbf{W}_i = D + \frac{s}{1-\rho} \sum_{i=1}^{n} \frac{\rho_i^2}{k_i} - \sum_{i=1}^{n} \frac{\rho_i(1-\rho_i) g_i^{(2)}}{2\lambda_i k_i}, \quad (6.1)$$

with

$$D = \rho \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}}{2(1-\rho)} + \rho \frac{s^{(2)}}{2s} + \frac{s}{2(1-\rho)} \left[\rho^2 - \sum_{i=1}^{n} \rho_i^2\right], \quad (6.2)$$

and $g_i^{(2)} = \mathrm{E}\mathbf{M}_i(\mathbf{M}_i - 1)$ with $\mathbf{M}_i$ denoting the number of customers served during a visit to Q_i. So for $k_i = 1$, $g_i^{(2)} = 0$, but for $k_i \neq 1$, $g_i^{(2)}$ is not known exactly.

6.3 The constrained optimization problem

In this section we study the problem of finding the service limits $k_1, \ldots, k_n$, constrained to $\sum_{i=1}^{n} \gamma_i k_i \leq K$, that minimize the mean total waiting cost $\sum_{i=1}^{n} c_i \lambda_i \mathrm{E}\mathbf{W}_i$. We successively consider the following four waiting-time approximations:

I. an approximation based on a 1-limited polling table;
II. a simple k-limited approximation;
III. a Fuhrmann & Wang-like k-limited approximation;
IV. the original Fuhrmann & Wang k-limited approximation.

In fact the approximations ignore the integrality of $k_1, \ldots, k_n$. The numbers $k_1, \ldots, k_n$, not necessarily integers, that minimize the resulting approximations for $\sum_{i=1}^{n} c_i \lambda_i \mathrm{E}\mathbf{W}_i$ are denoted by $k_1^*, \ldots, k_n^*$. The integers $k_1, \ldots, k_n$ that minimize the resulting approximations for $\sum_{i=1}^{n} c_i \lambda_i \mathrm{E}\mathbf{W}_i$ are denoted by $\bar{k}_1^*, \ldots, \bar{k}_n^*$. As the approximations are rather smooth, $k_1^*, \ldots, k_n^*$ are likely to provide an accurate indication for $\bar{k}_1^*, \ldots, \bar{k}_n^*$.

I. An approximation based on a 1-limited polling table.
A generalization of the cyclic visit order considered so far is a fixed, generally non-cyclic, visit order. Such a visit order may be described in a *polling table*, which may contain $m_i \geq 1$ visits to Q_i. Our approximation idea is the following. There is some resemblance between adopting the k_i-limited service discipline at Q_i, visiting Q_i once, and adopting the 1-limited service discipline at Q_i, visiting Q_i k_i times; in either case the server is allowed to serve at most k_i customers in one 'cycle'. So the optimal visit numbers $m_1, \ldots, m_n$ for the 1-limited service discipline may provide an indication for $k_1^*, \ldots, k_n^*$.
Boxma, Levy, & Weststrate [48] study the problem of finding those polling table visit numbers $m_1, \ldots, m_n$, that minimize $\sum_{i=1}^{n} c_i \lambda_i \mathbf{EW}_i$. They propose the following waiting-time approximation, under the assumption that the m_i visits to Q_i are spaced as evenly as possible:

$$\mathbf{EW}_i \approx A \frac{1-\rho+\rho_i}{1-\rho-\lambda_i \dfrac{\sum m_j s_j}{m_i}} \frac{\mathbf{EC}}{m_i}, \qquad i = 1, \ldots, n, \tag{6.3}$$

with $\mathbf{EC} = \sum_{j=1}^{n} m_j s_j/(1-\rho)$ the mean cycle time, and A some unknown constant. The constant A could be determined using the pseudo-conservation law for polling tables, cf. [45], but its value is not relevant for the determination of the optimal values of m_i, which follow easily from (6.3):

$$m_i^* = \lambda_i R + (1-\rho-\sum_{j=1}^{n} \lambda_j s_j) R \frac{\sqrt{c_i \lambda_i (1-\rho+\rho_i)/s_i}}{\sum_{j=1}^{n} s_j \sqrt{c_j \lambda_j (1-\rho+\rho_j)/s_j}}. \tag{6.4}$$

Here R represents an arbitrary scaling factor, reflecting the homogeneity of the objective function in $m_1, \ldots, m_n$.
As remarked before, the optimal visit numbers $m_1, \ldots, m_n$ for the 1-limited service discipline may provide an indication for $k_1^*, \ldots, k_n^*$. However, visiting Q_i k_i times in a cycle differs from visiting Q_i only once in the respect of the switch-over time incurred. In the former (latter) case the switch-over time corresponding to Q_i is incurred k_i times (once) per cycle. So the optimal visit numbers $m_1, \ldots, m_n$ may be better candidates to provide an indication for $k_1^*, \ldots, k_n^*$, when the mean switch-over times in (6.3) are scaled by a factor $1/m_i$; this gives

$$\mathbf{EW}_i \approx A \frac{1-\rho+\rho_i}{1-\rho-\lambda_i \dfrac{s}{m_i}} \frac{\mathbf{EC}}{m_i}, \qquad i = 1, \ldots, n, \tag{6.5}$$

with $\mathbf{EC} = s/(1-\rho)$. This leads to

$$m_i^* = \frac{\lambda_i s}{1-\rho} + (K - \sum_{j=1}^{n} \gamma_j \frac{\lambda_j s}{1-\rho}) \frac{\sqrt{c_i \lambda_i (1-\rho+\rho_i)/\gamma_i}}{\sum_{j=1}^{n} \gamma_j \sqrt{c_j \lambda_j (1-\rho+\rho_j)/\gamma_j}}. \tag{6.6}$$

One may interpret (6.6) as follows. The server should be allowed to serve at least $\frac{\lambda_i s}{1-\rho}$ customers during a visit to Q_i, to satisfy the stability condition. The remaining service capacity, $K - \sum_{j=1}^{n} \gamma_j \frac{\lambda_j s}{1-\rho}$, should be assigned proportionally to $\sqrt{c_i \lambda_i (1-\rho+\rho_i)/\gamma_i}$. Some reflection convinces one that indeed a station with relatively high c_i, λ_i, ρ_i, or $1/\gamma_i$ should be assigned a relatively high capacity.

Rule (6.6) is just as simple as (6.4) but yields better results. Still, the numerical results in Section 6.4 reveal that it does not always perform well. Below we investigate a quite different idea.

II. A simple k-limited approximation.
We now imitate the derivation of the waiting-time approximation for cyclic polling systems with 1-limited service [52] and for polling tables with 1-limited service ([48], leading to (6.3)). The waiting time of a (tagged) type-i customer is composed of:
i. the time from its arrival to the start of the subsequent visit of the server to Q_i, i.e., a residual cycle time $\mathbf{RC}_i$ with regard to Q_i;
ii. the time from the start of the latter visit to its service, i.e., approximately $\mathbf{X}_i/k_i$ cycle times $\mathbf{C}_i^+$ with regard to Q_i (atypical cycles, as each contains k_i services at Q_i), when the (tagged) customer finds $\mathbf{X}_i$ waiting type-i customers upon arrival.
Applying a traffic balance argument, $\mathrm{E}\mathbf{C}_i^+ \approx k_i\beta_i + s + (\rho-\rho_i)\mathrm{E}\mathbf{C}_i^+$. Noting that $\mathrm{E}\mathbf{X}_i = \lambda_i \mathrm{E}\mathbf{W}_i$, we thus obtain

$$\mathrm{E}\mathbf{W}_i \approx \frac{1-\rho+\rho_i}{1-\rho-\frac{\lambda_i}{k_i}s}\mathrm{E}\mathbf{RC}_i, \qquad i=1,\ldots,n. \tag{6.7}$$

For $k_i = 1$, (6.7) reduces to $\mathrm{E}\mathbf{W}_i \approx \frac{1-\rho+\rho_i}{1-\rho-\lambda_i s}\mathrm{E}\mathbf{RC}_i$, the known approximation [52] for the 1-limited service discipline. However, for $k_i = \infty$, (6.7) reduces to $\mathrm{E}\mathbf{W}_i \approx \frac{1-\rho+\rho_i}{1-\rho}\mathrm{E}\mathbf{RC}_i$, rather than $\mathrm{E}\mathbf{W}_i = (1-\rho_i)\mathrm{E}\mathbf{RC}_i$, the known exact result for the exhaustive service discipline (defining a cycle with regard to Q_i as the interval between two successive departures of S from Q_i). The reason for this discrepancy is that the derivation of (6.7) ignores the possibility that a type-i customer upon arrival finds S visiting Q_i and can still be served during that visit.

Starting from (6.7), assuming $\mathbf{ERC}_i \approx \mathbf{ERC} = B\,\mathbf{EC} = Bs/(1-\rho)$ with B some unknown constant,

$$k_i^* = \frac{\lambda_i s}{1-\rho} + (K - \sum_{j=1}^{n} \gamma_j \frac{\lambda_j s}{1-\rho}) \frac{\lambda_i \sqrt{c_i(1-\rho+\rho_i)/\gamma_i}}{\sum_{j=1}^{n} \gamma_j \lambda_j \sqrt{c_j(1-\rho+\rho_j)/\gamma_j}}. \tag{6.8}$$

One may interpret (6.8) similarly to (6.6).
Note that (6.8) slightly differs from (6.6) in the proportional assignment of the remaining service capacity, $K - \sum_{j=1}^{n} \gamma_j \frac{\lambda_j s}{1-\rho}$, which may be explained as follows. Visiting Q_i k_i times differs from visiting Q_i only once not only in the respect of the switch-over time incurred, as remarked before, but also differs in the respect of the residual time until visiting Q_i. In the former case the residual subcycle time is assumed to behave inversely proportional to k_i, whereas in the latter case the residual cycle time is assumed to behave constantly.

III. A Fuhrmann & Wang-like k-limited approximation.
To remedy the weakness of (6.7) indicated above, a natural heuristic approach is to take a weighted sum of the 1-limited waiting-time approximation $\mathbf{EW}_i \approx \frac{1-\rho+\rho_i}{1-\rho-\lambda_i s}\mathbf{ERC}_i$ and the exhaustive mean waiting-time result $\mathbf{EW}_i = (1-\rho_i)\mathbf{ERC}_i$, with weight factors $u_i(k_i)$ and $1-u_i(k_i)$, respectively. The choice $u_i(k_i) = \frac{1-\rho-\lambda_i s}{k_i(1-\rho)-\lambda_i s}$ has the desirable properties that $u_i(1) = 1$, $u_i(\infty) = 0$, and $\mathbf{EW}_i \to \infty$ for $k_i \to \frac{\lambda_i s}{1-\rho}$.
This in facts yields the approximation (30) of Fuhrmann & Wang [104]:

$$\mathbf{EW}_i \approx \frac{(1-\rho_i)(1-\rho) + \frac{\rho_i}{k_i}(2-\rho)}{1-\rho-\frac{\lambda_i s}{k_i}} \mathbf{ERC}_i, \qquad i = 1, \ldots, n. \tag{6.9}$$

Starting from (6.9), assuming $\mathbf{ERC}_i \approx \mathbf{ERC} = B\,\mathbf{EC}$, with B some unknown constant,

$$k_i^* = \frac{\lambda_i s}{1-\rho} + (K - \sum_{j=1}^{n} \gamma_j \frac{\lambda_j s}{1-\rho}) \frac{\sqrt{c_i \lambda_i \delta_i/\gamma_i}}{\sum_{j=1}^{n} \gamma_j \sqrt{c_j \lambda_j \delta_j/\gamma_j}}, \tag{6.10}$$

with $\delta_i = \rho_i(2-\rho) + \lambda_i s(1-\rho_i)$. One may interpret (6.10) similarly to (6.6).

IV. The original Fuhrmann & Wang k-limited approximation.
Fuhrmann & Wang [104] also assume $\mathbf{ERC}_i \approx \mathbf{ERC}$ but they do not assume $\mathbf{ERC} = B\mathbf{EC}$. Instead they approximate $\mathbf{ERC}$ by substituting (6.9) into (6.1), taking $g_i^{(2)} = 0$,

$$\mathbf{ERC} \approx \frac{D + \frac{s}{1-\rho}\sum_{j=1}^{n}\frac{\rho_j^2}{k_j}}{\sum_{j=1}^{n}[\rho_j(1-\rho_j) + \frac{\rho_j^2}{k_j}\frac{2-\rho}{1-\rho}]}. \tag{6.11}$$

Remember that $g_i^{(2)} = \mathbf{EM}_i(\mathbf{M}_i - 1) \geq (\mathbf{EM}_i)^2 - \mathbf{EM}_i = \left(\frac{\lambda_i s}{1-\rho}\right)^2 - \frac{\lambda_i s}{1-\rho}$.
Taking $g_i^{(2)} = \max\{0, \left(\frac{\lambda_i s}{1-\rho}\right)^2 - \frac{\lambda_i s}{1-\rho}\}$ in (6.1) would probably improve the accuracy of (6.11). We did however not consider this option, as the numerical results in Section 6.4 reveal that the rule based on (6.11) performs already very well.
Substituting (6.11) back into (6.9),

$$\mathbf{EW}_i \approx \frac{(1-\rho_i)(1-\rho) + \frac{\rho_i}{k_i}(2-\rho)}{1-\rho-\frac{\lambda_i s}{k_i}} \frac{D + \frac{s}{1-\rho}\sum_{j=1}^{n}\frac{\rho_j^2}{k_j}}{\sum_{j=1}^{n}[\rho_j(1-\rho_j) + \frac{\rho_j^2}{k_j}\frac{2-\rho}{1-\rho}]}. \tag{6.12}$$

The optimal service limits based on (6.12) can not be solved analytically but can easily be determined numerically.

In the next section we test the simple rules (6.6), (6.8), (6.10), and the rule based on (6.12).

Remark 6.3.1
Fuhrmann & Wang [104] concentrate on k-limited service under a *gated* regime at all queues. They use the reasoning leading to our approximation II, observe the discrepancy for $k_i = \infty$ (the reason for which is explained above (6.8)) and then modify their approximation in a way that amounts to our taking a weighted sum. Tedijanto [181] considers cyclic polling systems with a Bernoulli service policy. He proposes a waiting-time approximation which coincides with (6.9) when one replaces the Bernoulli parameters q_i by $1 - 1/k_i$. His approximation is used by Blanc & Van der Mei [23] to find those q_i that minimize a weighted sum of the mean waiting times, cf. Section 6.1.

□

Remark 6.3.2
Setting $\gamma_i = \beta_i$, $K = L - s$, i.e., imposing a limit L on the mean cycle time at periods of overload (namely, when all queues are loaded), (6.10) reduces to

$$k_i^* \beta_i = \frac{\rho_i s}{1-\rho} + (L - \frac{s}{1-\rho}) \frac{\sqrt{c_i \rho_i \delta_i}}{\sum_{j=1}^{n} \sqrt{c_j \rho_j \delta_j}}. \tag{6.13}$$

with $\delta_i = \rho_i(2-\rho) + \lambda_i s(1-\rho_i)$. One may interpret (6.13) as follows. The server should be allowed to visit Q_i at least for a time $\frac{\rho_i s}{1-\rho}$, to satisfy the stability condition. The remaining non-switch-over time, $L - \frac{s}{1-\rho}$, should be assigned proportionally to $\sqrt{c_i \rho_i \delta_i}$. This suggests a rule for the optimal setting of time-limits in polling models with a time-limited service discipline. Note that in the case of constant service times, the k-limited and time-limited service disciplines coincide.

□

6.4 Numerical results for the constrained problem

In this section we give an overview of the numerical results that we gathered to test the rules (6.6), (6.8), (6.10), and the rule based on (6.12) proposed in the previous section. For a wide variety of cases we compared the optimal values of the service limits and the waiting cost with the values achieved by the proposed rules.
To evaluate the mean waiting times we used the power-series algorithm (PSA). The PSA allows an accurate numerical determination of the mean waiting times in polling models for which the joint queue length process has the structure of a multi-dimensional quasi birth-death process, cf. [21], [22].
The main drawback from which the PSA suffers is that the time and memory requirements grow exponentially with the number of queues. We therefore confined ourselves to cases with only a few queues. We have confidence however that the various approaches will perform at least as good for a larger number of queues. In Section 6.5 we shall discuss the case of large n in some detail.
A further drawback from which the PSA suffers is that the time and memory requirements grow rapidly with the number of stages of the service and switch-over time distributions. For this reason, most of the numerical tests are conducted for cases with exponential service and switch-over times. The following arguments support our belief that the results for other service and switch-over time distributions will be similar in general. It should be noted that the k_i-values prescribed by the rules (6.6), (6.8), and (6.10) are insensitive to the form of the service time and switch-over time distributions. Indeed the form of the pseudo-conservation law, and of the waiting-time approximations based upon it, suggests that the optimal policy and optimal cost will depend

on higher service time moments mainly through $\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}$, the influence of individual service time distributions being marginal, and that the influence of the switch-over time distributions on the optimal policy will be very minor. Numerical experiments of Blanc & Van der Mei [23] for cyclic polling with a Bernoulli service policy (cf. Section 6.1) support this view completely. Our tests with an Erlang-2 service time distribution (cf. Table 6.6 of Section 6.6) also point in the direction of a robustness of the optimal policy w.r.t. service time distributions; a robustness that was also observed in designing optimal polling tables, cf. [48] pp. 152, 153, and 161. Finally it should be observed that Fuhrmann & Wang [104] p. 50 test their approximation only for exponentially distributed service times, but state that 'limited experience indicates that the accuracy for other service time distributions seems to be similar in general'.

The numerical results are presented in Tables 6.1-6.3. Table 6.1 contains 13 two-queue cases, Table 6.2 contains 2 three-queue cases, and Table 6.3 presents a five-queue case. Most of the parameter combinations are taken from [48]. In the two-queue cases we imposed the constraint $k_1 + k_2 \leq 12$, in the three-queue cases $k_1 + k_2 + k_3 \leq 12$, and in the five-queue case $k_1 + k_2 + \ldots + k_5 \leq 20$. The constraint may be interpreted as a limit on the maximal number of services in a cycle. The displayed cost figures are the *'exact'* waiting-cost figures obtained from the PSA.

$\lambda_1 = \lambda_2 = 0.75$; $\beta_1 = \beta_2 = 0.1$; $s_1 = s_2 = 0.1$; $\rho = 0.15$.

	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	(10, 2)	0.145	(9, 3)	0.146	(9, 3)	0.146	(10, 2)	0.145	0.0
(1, 0.2)	(8, 4)	0.159	(8, 4)	0.159	(8, 4)	0.159	(9, 3)	0.159	0.0
(1, 0.5)	(6, 6)	0.199	(7, 5)	0.199	(7, 5)	0.199	(7, 5)	0.199	0.0
(1, 1)	(6, 6)	0.265	(6, 6)	0.265	(6, 6)	0.265	(6, 6)	0.265	0.0
(1, 2)	(6, 6)	0.397	(5, 7)	0.397	(5, 7)	0.397	(5, 7)	0.397	0.0
(1, 5)	(4, 8)	0.794	(4, 8)	0.794	(4, 8)	0.794	(3, 9)	0.794	0.0
(1, 10)	(2, 10)	1.455	(3, 9)	1.455	(3, 9)	1.455	(2, 10)	1.455	0.0

$\lambda_1 = \lambda_2 = 0.75$; $\beta_1 = 0.9$; $\beta_2 = 0.1$; $s_1 = s_2 = 0.1$; $\rho = 0.75$.

	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(0.9, 0.1)	(8, 4)	2.169	(10, 2)	2.307	(9, 3)	2.209	(8, 4)	2.169	0.0
(1, 0.1)	(8, 4)	2.355	(10, 2)	2.476	(9, 3)	2.386	(9, 3)	2.386	1.3
(1, 0.2)	(6, 6)	2.676	(9, 3)	3.006	(8, 4)	2.842	(7, 5)	2.738	2.3
(1, 0.5)	(4, 8)	3.176	(8, 4)	4.301	(7, 5)	3.819	(4, 8)	3.176	0.0
(1, 2)	(3, 9)	4.972	(6, 6)	6.996	(6, 6)	6.996	(2, 10)	5.033	1.2
(1, 5)	(2, 10)	7.693	(5, 7)	10.99	(4, 8)	9.391	(2, 10)	7.693	0.0
(1, 10)	(2, 10)	12.13	(4, 8)	16.30	(4, 8)	16.30	(1, 11)	12.80	5.5

$\lambda_1 = \lambda_2 = 0.5$; $\beta_1 = 0.9$; $\beta_2 = 0.1$; $s_1 = s_2 = 0.1$; $\rho = 0.5$.

	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(0.9, 0.1)	(8, 4)	0.502	(9, 3)	0.502	(9, 3)	0.502	(9, 3)	0.502	0.0
(1, 0.1)	(9, 3)	0.548	(10, 2)	0.550	(10, 2)	0.550	(9, 3)	0.548	0.0
(1, 0.2)	(6, 6)	0.620	(9, 3)	0.632	(9, 3)	0.632	(7, 5)	0.623	0.5
(1, 0.5)	(3, 9)	0.780	(8, 4)	0.861	(8, 4)	0.861	(3, 9)	0.780	0.0
(1, 2)	(1, 11)	1.370	(6, 6)	1.804	(6, 6)	1.804	(1, 11)	1.370	0.0
(1, 5)	(1, 11)	2.292	(4, 8)	3.229	(4, 8)	3.229	(1, 11)	2.292	0.0
(1, 10)	(1, 11)	3.827	(4, 8)	5.942	(4, 8)	5.942	(1, 11)	3.827	0.0

$\lambda_1 = \lambda_2 = 0.4$; $\beta_1 = 0.9$; $\beta_2 = 0.1$; $s_1 = s_2 = 1.5$; $\rho = 0.4$.

	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 50)	(3, 9)	68.77	(4, 8)	71.78	(4, 8)	71.78	(3, 9)	68.77	0.0
(1, 10)	(4, 8)	15.92	(5, 7)	16.74	(5, 7)	16.74	(4, 8)	15.92	0.0
(1, 3)	(5, 7)	6.126	(6, 6)	6.305	(6, 6)	6.305	(5, 7)	6.126	0.0
(1, 1)	(6, 6)	3.037	(7, 5)	3.156	(7, 5)	3.156	(6, 6)	3.037	0.0
(3, 1)	(7, 5)	5.770	(8, 4)	5.968	(8, 4)	5.968	(7, 5)	5.770	0.0
(10, 1)	(8, 4)	14.66	(9, 3)	15.54	(9, 3)	15.54	(8, 4)	14.66	0.0
(50, 1)	(9, 3)	63.64	(9, 3)	63.64	(9, 3)	63.64	(9, 3)	63.64	0.0

$\lambda_1 = \lambda_2 = 0.8$; $\beta_1 = 0.9$; $\beta_2 = 0.1$; $s_1 = s_2 = 0.4$; $\rho = 0.8$.

	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 50)	(4, 8)	120.9	(5, 7)	132.5	(4, 8)	120.9	(4, 8)	120.9	0.0
(1, 10)	(5, 7)	34.19	(6, 6)	76.59	(5, 7)	34.19	(4, 8)	38.96	13.9
(1, 3)	(5, 7)	16.98	(7, 5)	41.70	(6, 6)	27.90	(5, 7)	16.98	0.0
(1, 1)	(5, 7)	12.06	(7, 5)	17.69	(7, 5)	17.69	(6, 6)	13.99	15.9
(3, 1)	(6, 6)	28.04	(8, 4)	39.79	(7, 5)	29.04	(7, 5)	29.04	3.6
(10, 1)	(7, 5)	68.78	(8, 4)	70.43	(8, 4)	70.43	(7, 5)	68.78	0.0
(50, 1)	(8, 4)	245.6	(8, 4)	245.6	(8, 4)	245.6	(8, 4)	245.6	0.0

$\lambda_1 = 0.765$; $\lambda_2 = 0.085$; $\beta_1 = \beta_2 = 0.1$; $s_1 = s_2 = 0.1$; $\rho = 0.085$.

	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	(10, 2)	0.124	(11, 1)	0.124	(11, 1)	0.124	(11, 1)	0.124	0.0
(1, 0.2)	(9, 3)	0.125	(10, 2)	0.125	(11, 1)	0.125	(11, 1)	0.125	0.0
(1, 0.5)	(9, 3)	0.130	(10, 2)	0.130	(11, 1)	0.130	(11, 1)	0.130	0.0
(1, 1)	(9, 3)	0.137	(9, 3)	0.137	(11, 1)	0.137	(11, 1)	0.137	0.0
(1, 2)	(8, 4)	0.151	(8, 4)	0.151	(10, 2)	0.151	(10, 2)	0.151	0.0
(1, 5)	(8, 4)	0.194	(7, 5)	0.194	(10, 2)	0.194	(9, 3)	0.194	0.0
(1, 10)	(8, 4)	0.266	(6, 6)	0.266	(9, 3)	0.266	(8, 4)	0.266	0.0

$\lambda_1 = 0.765$; $\lambda_2 = 0.085$; $\beta_1 = \beta_2 = 0.5$; $s_1 = s_2 = 0.1$; $\rho = 0.425$.

	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	(11, 1)	0.393	(11, 1)	0.393	(11, 1)	0.393	(11, 1)	0.393	0.0
(1, 0.2)	(11, 1)	0.400	(11, 1)	0.400	(11, 1)	0.400	(11, 1)	0.400	0.0
(1, 0.5)	(11, 1)	0.422	(11, 1)	0.422	(11, 1)	0.422	(11, 1)	0.422	0.0
(1, 1)	(10, 2)	0.456	(9, 3)	0.456	(11, 1)	0.458	(11, 1)	0.458	0.4
(1, 2)	(8, 4)	0.519	(9, 3)	0.519	(10, 2)	0.520	(9, 3)	0.519	0.0
(1, 5)	(4, 8)	0.684	(8, 4)	0.701	(9, 3)	0.706	(6, 6)	0.692	1.2
(1, 10)	(2, 10)	0.918	(8, 4)	1.006	(9, 3)	1.018	(2, 10)	0.918	0.0

$\lambda_1 = 0.765$; $\lambda_2 = 0.085$; $\beta_1 = \beta_2 = 1$; $s_1 = s_2 = 0.1$; $\rho = 0.85$.

	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	(11, 1)	4.333	(11, 1)	4.333	(11, 1)	4.333	(11, 1)	4.333	0.0
(1, 0.2)	(11, 1)	4.466	(11, 1)	4.466	(11, 1)	4.466	(11, 1)	4.466	0.0
(1, 0.5)	(11, 1)	4.865	(11, 1)	4.865	(11, 1)	4.865	(11, 1)	4.865	0.0
(1, 1)	(11, 1)	5.529	(10, 2)	5.565	(11, 1)	5.529	(10, 2)	5.565	6.8
(1, 2)	(9, 3)	6.013	(10, 2)	6.090	(10, 2)	6.090	(9, 3)	6.013	0.0
(1, 5)	(7, 5)	7.047	(9, 3)	7.154	(9, 3)	7.154	(6, 6)	7.114	1.0
(1, 10)	(6, 6)	8.428	(8, 4)	8.692	(9, 3)	9.055	(4, 8)	8.630	2.4

$\lambda_1 = 0.765$; $\lambda_2 = 0.085$; $\beta_1 = \beta_2 = 0.5$; $s_1 = s_2 = 0.4$; $\rho = 0.425$.

	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 50)	(3, 9)	5.899	(5, 7)	6.174	(7, 5)	6.441	(3, 9)	5.899	0.0
(1, 10)	(6, 6)	1.932	(7, 5)	1.934	(9, 3)	1.950	(8, 4)	1.941	0.5
(1, 3)	(9, 3)	1.127	(9, 3)	1.127	(10, 2)	1.133	(10, 2)	1.133	0.5
(1, 1)	(10, 2)	0.888	(10, 2)	0.888	(11, 1)	0.905	(11, 1)	0.905	1.9
(3, 1)	(11, 1)	2.413	(11, 1)	2.413	(11, 1)	2.413	(11, 1)	2.413	0.0
(10, 1)	(11, 1)	7.690	(11, 1)	7.690	(11, 1)	7.690	(11, 1)	7.690	0.0
(50, 1)	(11, 1)	37.85	(11, 1)	37.85	(11, 1)	37.85	(11, 1)	37.85	0.0

$\lambda_1 = 0.765$; $\lambda_2 = 0.085$; $\beta_1 = \beta_2 = 1$; $s_1 = s_2 = 0.4$; $\rho = 0.85$.									
	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 50)	(8, 4)	28.52	(8, 4)	28.52	(8, 4)	28.52	(7, 5)	30.33	6.3
(1, 10)	(10, 2)	12.65	(9, 3)	12.81	(10, 2)	12.65	(9, 3)	12.81	1.2
(1, 3)	(10, 2)	9.20	(10, 2)	9.20	(11, 1)	9.61	(10, 2)	9.20	0.0
(1, 1)	(11, 1)	4.92	(11, 1)	4.92	(11, 1)	4.92	(11, 1)	4.92	0.0
(3, 1)	(11, 1)	10.08	(11, 1)	10.08	(11, 1)	10.08	(11, 1)	10.08	0.0
(10, 1)	(11, 1)	28.11	(11, 1)	28.11	(11, 1)	28.11	(11, 1)	28.11	0.0
(50, 1)	(11, 1)	131.2	(11, 1)	131.2	(11, 1)	131.2	(11, 1)	131.2	0.0

$\lambda_1 = 0.765$; $\lambda_2 = 0.085$; $\beta_1 = \beta_2 = 1$; $s_1 = 0.1$; $s_2 = 0.7$; $\rho = 0.85$.									
	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 50)	(8, 4)	29.60	(8, 4)	29.60	(8, 4)	29.60	(7, 5)	30.33	2.5
(1, 10)	(10, 2)	13.00	(9, 3)	13.14	(10, 2)	13.00	(9, 3)	13.14	1.0
(1, 3)	(10, 2)	9.413	(10, 2)	9.413	(11, 1)	9.857	(11, 1)	9.857	4.7
(1, 1)	(11, 1)	5.060	(11, 1)	5.060	(11, 1)	5.060	(11, 1)	5.060	0.0
(3, 1)	(11, 1)	10.38	(11, 1)	10.38	(11, 1)	10.38	(11, 1)	10.38	0.0
(10, 1)	(11, 1)	29.01	(11, 1)	29.01	(11, 1)	29.01	(11, 1)	29.01	0.0
(50, 1)	(11, 1)	135.5	(11, 1)	135.5	(11, 1)	135.5	(11, 1)	135.5	0.0

$\lambda_1 = 0.5$; $\lambda_2 = 0.25$; $\beta_1 = \beta_2 = 1$; $s_1 = 0.1$; $s_2 = 0.2$; $\rho = 0.75$.									
	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 18.4)	(2, 10)	10.66	(4, 8)	12.54	(4, 8)	12.54	(1, 11)	12.30	15.4
(1, 7.6)	(3, 9)	6.278	(5, 7)	6.885	(6, 6)	7.434	(2, 10)	6.293	0.2
(1, 6)	(3, 9)	5.515	(6, 6)	6.286	(6, 6)	6.286	(2, 10)	5.646	2.4
(1, 3.95)	(3, 9)	4.537	(6, 6)	4.813	(6, 6)	4.813	(3, 9)	4.537	0.0
(1, 2.75)	(5, 7)	3.863	(7, 5)	4.198	(7, 5)	4.198	(4, 8)	3.882	0.5
(1, 1.5)	(6, 6)	3.036	(8, 4)	3.280	(8, 4)	3.280	(6, 6)	3.036	0.0
(1, 0.7)	(10, 2)	2.265	(9, 3)	2.294	(9, 3)	2.294	(9, 3)	2.294	1.3

$\lambda_1 = 0.5$; $\lambda_2 = 1$; $\beta_1 = 1$; $\beta_2 = 0.3$; $s_1 = 0.2$; $s_2 = 0.6$; $\rho = 0.8$.									
	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 50)	(3, 9)	129.7	(3, 9)	129.7	(3, 9)	129.7	(3, 9)	129.7	0.0
(1, 10)	(3, 9)	32.09	(4, 8)	69.06	(4, 8)	69.06	(3, 9)	32.09	0.0
(1, 3)	(3, 9)	15.01	(4, 8)	23.77	(4, 8)	23.77	(3, 9)	15.01	0.0
(1, 1)	(3, 9)	10.13	(5, 7)	12.18	(5, 7)	12.18	(4, 8)	10.83	6.9
(3, 1)	(5, 7)	17.53	(6, 6)	19.61	(6, 6)	19.61	(5, 7)	17.53	0.0
(10, 1)	(6, 6)	32.79	(7, 5)	41.42	(7, 5)	41.42	(6, 6)	32.79	0.0
(50, 1)	(7, 5)	99.50	(7, 5)	99.50	(7, 5)	99.50	(7, 5)	99.50	0.0

TABLE 6.1. The constrained case; two-queue models.

$\lambda_1 = \lambda_2 = \lambda_3 = 0.7$; $\beta_1 = 0.8$; $\beta_2 = \beta_3 = 0.1$; $s_1 = s_2 = s_3 = 0.05$; $\rho = 0.7$.									
	optimum		(6.8)		(6.10)		(6.12)		
$(c_1, c_{2,3})$	$(k_{1,2,3})$	cost	$(k_{1,2,3})$	cost	$(k_{1,2,3})$	cost	$(k_{1,2,3})$	cost	%
(3.5, 1)	(3, 4, 5)	7.004	(7, 2, 3)	8.547	(6, 3, 3)	7.717	(4, 4, 4)	7.198	2.8
(14.4, 1)	(9, 2, 1)	18.09	(8, 2, 2)	18.20	(8, 2, 2)	18.20	(8, 2, 2)	18.20	0.6
(51.8, 1)	(9, 2, 1)	50.50	(10, 1, 1)	51.80	(9, 1, 2)	52.07	(10, 1, 1)	51.80	2.6

$\lambda_1 = 0.531$; $\lambda_2 = 0.212$; $\lambda_3 = 0.106$; $\beta_1 = \beta_2 = \beta_3 = 0.9$; $s_1 = s_2 = s_3 = 0.3$; $\rho = 0.7641$.									
	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_2, c_3)	$(k_{1,2,3})$	cost	$(k_{1,2,3})$	cost	$(k_{1,2,3})$	cost	$(k_{1,2,3})$	cost	%
(1, 0.1, 1)	(9, 1, 2)	2.660	(8, 2, 2)	2.668	(8, 2, 2)	2.668	(8, 2, 2)	2.668	0.3
(1, 5, 1)	(5, 5, 2)	7.796	(6, 5, 1)	8.541	(6, 5, 1)	8.541	(5, 6, 1)	8.055	3.3
(1, 1, 5)	(6, 3, 3)	6.252	(6, 3, 3)	6.252	(6, 3, 3)	6.252	(6, 3, 3)	6.252	0.0

TABLE 6.2. The constrained case; three-queue models.

$\lambda_1 = 0.35$; $\lambda_2 = \ldots = \lambda_5 = 0.1$; $\beta_1 = 1$; $\beta_2 = \ldots = \beta_5 = 1$; $s_1 = 0.1$; $s_2 = \ldots = s_5 = 0.05$; $\rho = 0.75$.									
	optimum		(6.8)		(6.10)		(6.12)		
(c_1, c_{2-5})	(k_1, k_{2-5})	cost	(k_1, k_{2-5})	cost	(k_1, k_{2-5})	cost	(k_1, k_{2-5})	cost	%
(1, 0.1)	(16, 1)	0.883	(16, 1)	0.883	(16, 1)	0.883	(16, 1)	0.883	0.0
(1, 0.5)	(16, 1)	1.804	(8, 3)	1.987	(12, 2)	1.807	(16, 1)	1.804	0.0
(1, 1)	(4, 4)	2.420	(8, 3)	2.934	(8, 3)	2.934	(8, 3)	2.934	21.6
(1, 2)	(4, 4)	3.631	(4, 4)	3.631	(8, 3)	4.827	(4, 4)	3.631	0.0

TABLE 6.3. The constrained case; a five-queue model.

Discussion of the numerical results.
The various rules perform reasonably well; in the majority of the examples the waiting cost achieved is less than 10% larger than the minimal waiting cost. It is however interesting to compare how the various rules perform. For compactness we did not display the results for the rule (6.6), but on average the rules (6.6) and (6.8) perform similarly. Sometimes (6.6) performs better, sometimes (6.8). The rule (6.10) performs slightly better than the rule (6.8). The underlying approximation of (6.10) is theoretically indeed better than the underlying approximation of (6.8). The former shows the correct exact behavior when $k_i \to \infty$, the latter does not. As the imposed constraint prevents that $k_i \to \infty$, the difference in performance is however minor.
The rule based on the approximation (6.12) performs by far the best; only 8 out of the 101 times the relative error exceeds 5%. The approximation (6.12) is indeed theoretically better than the underlying approximation of (6.10). The former catches the influence of k_i on EW_j, the latter does not. Thus the difference in performance will especially be dramatic in cases where that influence plays a crucial role, as is illustrated by the numerical results. When we take a closer look at those two-queue cases where the rules (6.6), (6.8), and (6.10) perform poorly, we notice that simultaneously β_1 is larger than β_2 and c_1 is smaller than c_2. These rules, which completely ignore the influence of k_1 on EW_2, then choose k_1 too large and k_2 too small. These two-queue cases are typically cases where the influence of k_1 on EW_2 plays a crucial role: for large β_1 the influence of k_1 on EW_2 is large and for large c_2 this influence is heavily weighted in the waiting cost.
Concluding, when very high accuracy is not needed, we recommend to use the simple rule (6.10); otherwise the rule based on (6.12) should be used. By its simple explicit form the rule (6.10) may also serve to provide some insight in the influence of the various parameters.

6.5 THE UNCONSTRAINED OPTIMIZATION PROBLEM

In this section we study the problem of finding the (unconstrained) service limits $k_1, \ldots, k_n$ that minimize the waiting cost $\sum_{i=1}^{n} c_i \lambda_i \mathrm{EW}_i$. We first derive several monotonicity properties of polling systems with k-limited service and switch-over periods. The main result is a (partially conjectured) rule stating that for minimizing the waiting cost in such systems the queues with the highest value of c_i/β_i should be assigned $k_i = \infty$, i.e., receive exhaustive service.

This property is very similar to the well-known $c\mu$-rule derived for systems with no switch-over periods and in which the server is free to move from queue to queue dynamically. We subsequently propose to use the Fuhrmann & Wang approximation for the unconstrained waiting-cost minimization. We specifically investigate to what extent the Fuhrmann & Wang approximation satisfies the above-mentioned properties.

Proposition 6.5.1
In a stable polling system with cyclic visit order and k-limited service the sum $\sum_{j=1}^{n} \rho_j \mathrm{E}\mathbf{W}_j$ is non-increasing in each of the service limits k_i, $i = 1, \ldots, n$.

Proof
Let $\mathbf{V}_t$ be the total amount of work in the system at time t. Let $\mathbf{V}$ be a stochastic variable with distribution the steady-state distribution of the total amount of work in the system. As shown in [142] $\mathbf{V}_t$ is non-increasing in k_i (the proof in [142] is a path-wise proof). Hence $\mathrm{E}\mathbf{V}$ is non-increasing in k_i. Now, it is known that

$$\mathrm{E}\mathbf{V} = \sum_{j=1}^{n} \rho_j \mathrm{E}\mathbf{W}_j + \sum_{j=1}^{n} \rho_j \frac{\beta_j^{(2)}}{2\beta_j}$$

and thus $\sum_{j=1}^{n} \rho_j \mathrm{E}\mathbf{W}_j$ is non-increasing in k_i.

□

Conjecture 6.5.1
In a stable polling system with cyclic visit order and k-limited service the mean waiting time at Q_i, $\mathrm{E}\mathbf{W}_i$, is decreasing in its service limit k_i and increasing in k_j for every $j \neq i$.

While the claim made in Conjecture 6.5.1 is very appealing and intuitive, it seems difficult to prove it. A reasonable line of argument can nonetheless be provided as follows. To see the effect of k_j on $\mathrm{E}\mathbf{W}_i$ one can view the services given at Q_j as switch-over periods (whose durations are distributed as the sum of several independent random variables, whose number is the number of customers being served at Q_j). It is easy to see that as long as the system is stable the mean number of services given at Q_j per visit is constant ($\frac{\lambda_j s}{1-\rho}$, which does not depend on k_j). On the other hand, the second moment of the number of services *does* depend on k_j. Increasing k_j will increase the number of services given at Q_j when that queue is loaded (has more than k_j customers), and thus is likely to decrease the number of services given at Q_j when it has less than k_j customers. This implies that the variance of the number of services

given at Q_j increases with k_j. Going back to the viewpoint of Q_i, we see that when k_j increases, the customers at Q_i observe switch-over periods of the same mean but of higher variance. Viewing Q_i as an $M/G/1$ queue with vacations (where the services of the other queues and the switch-over periods together constitute one large vacation), it is likely that increasing the vacation second moment while keeping its mean the same increases the mean waiting time at Q_i. The latter property is indeed known to hold for exhaustive and gated $M/G/1$ vacation queues, cf. Takagi [174]. We thus conclude that the mean waiting time at Q_i is increasing in k_j for any $j \neq i$. Combining this fact with Proposition 6.5.1 implies that the mean waiting time at Q_i must be decreasing in k_i.

Remark 6.5.1
The assumption that the system is stable plays an essential role in Conjecture 6.5.1. If Q_j is unstable, then increasing k_j will not only increase the variance of the intervisit time of Q_i, but also the mean. If the increment of the first moment is larger than the increment of the second moment, then the residual intervisit time of Q_i will *decrease*. The mean waiting time at Q_i will then also *decrease*.

□

Proposition 6.5.1 and Conjecture 6.5.1 lead to the main result of this section, namely that for optimality at least one of the queues, viz. the one whose c_i/β_i achieves the maximum value, must be served without limits.

Conjecture 6.5.2
If $c_i/\beta_i = \max_{j=1,\ldots,n} c_j/\beta_j$, then the optimal service limit $k_i^* = \infty$.

Proof
We need to show that for all $(k_1, \ldots, k_n)$ the sum $\sum_{j=1}^{n} c_j \lambda_j \mathrm{E}\mathbf{W}_j$ is decreasing in k_i, i.e., $\frac{\Delta}{\Delta k_i} \sum_{j=1}^{n} \lambda_j c_j \mathrm{E}\mathbf{W}_j \leq 0$ (with $\frac{\Delta}{\Delta k_i}$ denoting partial difference):

$$\begin{aligned}
\frac{\Delta}{\Delta k_i} \sum_{j=1}^{n} \lambda_j c_j \mathrm{E}\mathbf{W}_j &= \frac{c_i}{\beta_i} \frac{\Delta}{\Delta k_i} \rho_i \mathrm{E}\mathbf{W}_i + \sum_{j \neq i} \frac{c_j}{\beta_j} \frac{\Delta}{\Delta k_i} \rho_j \mathrm{E}\mathbf{W}_j \\
&\leq \frac{c_i}{\beta_i} \frac{\Delta}{\Delta k_i} \rho_i \mathrm{E}\mathbf{W}_i + \frac{c_i}{\beta_i} \sum_{j \neq i} \frac{\Delta}{\Delta k_i} \rho_j \mathrm{E}\mathbf{W}_j \\
&= \frac{c_i}{\beta_i} \frac{\Delta}{\Delta k_i} \sum_{j=1}^{n} \rho_j \mathrm{E}\mathbf{W}_j \leq 0.
\end{aligned}$$

The inequality in the second line follows from the facts that $\frac{\Delta}{\Delta k_i}\mathbf{EW}_j$ is non-negative (due to Conjecture 6.5.1) and that $c_i/\beta_i \geq c_j/\beta_j$ (condition of the theorem). The inequality in the third line follows from Proposition 6.5.1.

□

Remark 6.5.2
Conjecture 6.5.2 implies that if the service limit policies are of the limited-gated type (namely, serve up to k_i customers but only of those present at the queue at the polling instant), then the queues with the maximal value of c_i/β_i should be served according to the gated policy with $k_i^* = \infty$. If the service limit policies are of the exhaustive-limited type (namely serve up to k_i customers but allow to include in these services customers that arrived during the service of the queue), then the queues with the maximal value of c_i/β_i should be served according to the exhaustive policy with $k_i^* = \infty$. This reminds of the $c\mu$-rule derived for systems with no switch-over periods and in which the server is free to move from queue to queue. According to the $c\mu$-rule the queues with the highest value of c_i/β_i should receive the highest priority in the system, which implies, in particular, exhaustive service at those queues.

□

Keeping in mind the properties discussed above we now study the problem of finding the service limits $k_1, \ldots, k_n$ that minimize the waiting cost $\sum_{i=1}^{n} c_i\lambda_i\mathbf{EW}_i$.
We propose to use the Fuhrmann & Wang approximation (6.12). As observed in the previous section, for the constrained waiting-cost minimization the Fuhrmann & Wang approximation outperforms the simpler approximations. For the unconstrained waiting-cost minimization the simpler approximations would be useless anyhow, as they completely ignore the influence of k_j on $\mathbf{EW}_i$, which would always lead to $k_1^* = \infty, \ldots, k_n^* = \infty$.
We now specifically investigate to what extent the Fuhrmann & Wang approximation (6.12) satisfies the properties discussed above.

Proposition 6.5.2
The approximation (6.12) of $\mathbf{EW}_i$ is: i) decreasing in k_i, and ii) increasing in k_j, $j \neq i$.

Proof
A straightforward computation shows that $\mathbf{EW}_i \mid_{k_i=r} \geq \mathbf{EW}_i \mid_{k_i=r+1}$ and that $\mathbf{EW}_i \mid_{k_j=r} \leq \mathbf{EW}_i \mid_{k_j=r+1}$, $j \neq i$. (The latter inequality holds provided $D \geq \frac{s}{2-\rho}\sum_{j=1}^{n}\rho_j(1-\rho_j)$, which may be shown to hold by substitution of the definition of D.)

□

Proposition 6.5.2 supports the use of (6.12) in trying to obtain the optimal service limit values for the actual polling system. Moreover, in the numerical experiments that will be presented in the next section we will find that the minimal value of $\sum_{i=1}^{n} c_i \lambda_i \mathrm{E}\mathbf{W}_i$, where (6.12) is used to evaluate $\mathrm{E}\mathbf{W}_i$, is always achieved when the service limit of the queue with the maximal value of c_i/β_i is set to $k_i = \infty$ (exhaustive service).

In the numerical experiments for both the constrained case (Section 6.4) and the unconstrained case (Section 6.6), time and memory requirements of the PSA have forced us to confine ourselves to models with only a few queues. Let us now discuss what happens when the number of queues, n, approaches infinity, distinguishing four cases: for all j:

I. s_j fixed, $\beta_j = O(1/n)$, λ_j fixed;
II. $s_j = O(1/n)$, β_j fixed, $\lambda_j = O(1/n)$;
III. $s_j = O(1/n)$, $\beta_j = O(1/n)$, λ_j fixed;
IV. s_j fixed, β_j fixed, $\lambda_j = O(1/n)$.

In case I, $\lambda_i s/(1-\rho) \to \infty$ and hence necessarily $k_i \to \infty$; this is not an interesting case. Case II reduces to continuous polling on a circle; cf. Fuhrmann & Cooper [102]. Each customer will be served in the cycle in which it arrives, even if the k_i values equal one; the actual choice of the k_i is irrelevant. Cases III and IV are equivalent up to a scaling of time by a factor n. Let us discuss case III in more detail. For the constrained situation, (6.7), (6.9) and (6.12) all reduce to

$$\mathrm{E}\mathbf{W}_i \approx B \frac{1-\rho}{1-\rho-\dfrac{\lambda_i s}{k_i}}, \tag{6.14}$$

with B some constant, leading to

$$k_i^* = \frac{\lambda_i s}{1-\rho} + (K - \sum_{j=1}^{n} \gamma_j \frac{\lambda_j s}{1-\rho}) \frac{\lambda_i \sqrt{c_i/\gamma_i}}{\sum_{j=1}^{n} \gamma_j \lambda_j \sqrt{c_j/\gamma_j}}. \tag{6.15}$$

Note that the weakness of approximation (6.7), indicated above (6.8), disappears when $n \to \infty$. Approximation (6.14) may be expected to perform very well. For the unconstrained situation, (6.12) also reduces to (6.14). Hence the waiting cost is minimized by taking $k_i = \infty$ for all i. Indeed, for large finite n an increment of k_i by one reduces $c_i \lambda_i \mathrm{E}\mathbf{W}_i$ much stronger than it increases $\sum_{j \neq i} c_j \lambda_j \mathrm{E}\mathbf{W}_j$, as is indicated by the following rough reasoning.

To make things simple, let us assume that $k_2 = \ldots = k_n = \infty$; now increase k_1 by one. Customers in Q_1 only notice this increment at a server visit when at least $k_1 + 1$ customers are present. Suppose such an event occurs in the mth cycle. Now this saves one Q_1 customer one cycle time $\mathbf{C}_{1,m+1}$, which is

$O(1)$. Here $\mathbf{C}_{j,m}$ denotes the mth cycle time for Q_j. What is the effect on some other queue Q_j? First the bad effect. S reaches Q_j Δ_j later; this delay consists of a service time at Q_1 (of $O(1/n)$) and of extended visit times at $Q_2, \ldots, Q_{j-1}$; $\Delta_j = O(1/n)$. Each of the customers at Q_j experiences this additional delay as an addition to its waiting time. There are on the average $\lambda_j \mathrm{E}\mathbf{C}_{j,m}$ such customers. The total mean 'loss' for Q_j is $\Delta_j \lambda_j \mathrm{E}\mathbf{C}_{j,m}$. Here we ignore an $O(n^{-2})$ contribution: compared to an ordinary cycle, this one lasted already Δ_j longer, during which additional period also on the average $\lambda_j \Delta_j$ customers have arrived who each experience an extra delay Δ_j. Now there is also a *benefit* for Q_j. During the extra delay also customers arrive at Q_j who are just in time to be served in *this* cycle; without the extra delay they would have arrived just after the departure of S from Q_j and would have had to wait a full cycle. The total mean 'gain' for Q_j is $\Delta_j \lambda_j \mathrm{E}\mathbf{C}_{j,m+1} + O(n^{-2})$. The result on Q_j of these two counteracting effects is $\lambda_j \Delta_j (\mathrm{E}\mathbf{C}_{j,m} - \mathrm{E}\mathbf{C}_{j,m+1}) + O(n^{-2})$ (the propagation of an extra service in Q_1 in later cycles should also have an $O(n^{-2})$ effect). Obviously, $\mathrm{E}\mathbf{C}_{j,m} - \mathrm{E}\mathbf{C}_{j,m+1} = O(1)$, but it seems in fact likely that $\mathrm{E}\mathbf{C}_{j,m} - \mathrm{E}\mathbf{C}_{j,m+1} = O(1/n)$. In the latter case increasing k_1 by one has an $O(n^{-2})$ effect on Q_j, which agrees with (6.14).

6.6 Numerical results for the unconstrained problem

In this section we give an overview of the numerical results that we gathered to test the approach proposed in the previous section. For a wide variety of cases we compared the optimal values of the service limits and the waiting cost with the values achieved by the proposed approximative approach.
Just like in Section 6.4 we used the power-series algorithm (PSA) to evaluate the mean waiting times and we confined ourselves to cases with only a few queues. We further focused again on cases with an exponential service and switch-over time distribution, although we did investigate some cases with an Erlang-2 service time distribution as well. The results for an Erlang-2 service time distribution appear to be similar to the results for an exponential service time distribution.

The numerical results are presented in Tables 6.4-6.7. Table 6.4 contains the same 13 two-queue cases as Table 6.1 of Section 6.4, Table 6.5 contains 4 three-queue cases with exponential service time distributions, Table 6.6 contains the same cases but with Erlang-2 service time distributions with the same mean, and Table 6.7 contains the same five-queue case as Table 6.3 of Section 6.4. The displayed cost figures are the *'exact'* waiting cost figures obtained from the PSA.

$\lambda_1 = \lambda_2 = 0.75$; $\beta_1 = \beta_2 = 0.1$; $s_1 = s_2 = 0.1$; $\rho = 0.15$.

	optimum		approximation		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	$(\infty, 2)$	0.145	$(\infty, 3)$	0.145	0.0
(1, 0.2)	$(\infty, 5)$	0.159	(∞, ∞)	0.159	0.0
(1, 0.5)	$(\infty, 13)$	0.199	(∞, ∞)	0.199	0.0
(1, 1)	(∞, ∞)	0.265	(∞, ∞)	0.265	0.0
(1, 2)	$(13, \infty)$	0.397	(∞, ∞)	0.397	0.0
(1, 5)	$(5, \infty)$	0.794	(∞, ∞)	0.794	0.0
(1, 10)	$(2, \infty)$	1.455	$(3, \infty)$	1.455	0.0

$\lambda_1 = \lambda_2 = 0.75$; $\beta_1 = 0.9$; $\beta_2 = 0.1$; $s_1 = s_2 = 0.1$; $\rho = 0.75$.

	optimum		approximation		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	$(\infty, 17)$	2.158	(∞, ∞)	2.162	0.2
(1, 0.2)	$(12, \infty)$	2.535	$(20, \infty)$	2.558	0.9
(1, 0.5)	$(5, \infty)$	3.119	$(5, \infty)$	3.119	0.0
(1, 1)	$(4, \infty)$	3.817	$(3, \infty)$	3.836	0.5
(1, 2)	$(3, \infty)$	4.949	$(2, \infty)$	5.030	1.6
(1, 5)	$(2, \infty)$	7.685	$(2, \infty)$	7.685	0.0
(1, 10)	$(2, \infty)$	12.11	$(1, \infty)$	12.80	5.7

$\lambda_1 = \lambda_2 = 0.5$; $\beta_1 = 0.9$; $\beta_2 = 0.1$; $s_1 = s_2 = 0.1$; $\rho = 0.5$.

	optimum		approximation		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	(∞, ∞)	0.540	(∞, ∞)	0.540	0.0
(1, 0.2)	$(7, \infty)$	0.617	(∞, ∞)	0.623	1.0
(1, 0.5)	$(3, \infty)$	0.780	$(3, \infty)$	0.780	0.0
(1, 1)	$(2, \infty)$	0.992	$(2, \infty)$	0.992	0.0
(1, 2)	$(1, \infty)$	1.370	$(1, \infty)$	1.370	0.0
(1, 5)	$(1, \infty)$	2.292	$(1, \infty)$	2.292	0.0
(1, 10)	$(1, \infty)$	3.827	$(1, \infty)$	3.827	0.0

$\lambda_1 = \lambda_2 = 0.4$; $\beta_1 = 0.9$; $\beta_2 = 0.1$; $s_1 = s_2 = 1.5$; $\rho = 0.4$.

	optimum		approximation		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	(∞, ∞)	1.305	(∞, ∞)	1.305	0.0
(1, 0.2)	(∞, ∞)	1.469	(∞, ∞)	1.469	0.0
(1, 0.5)	$(20, \infty)$	1.963	(∞, ∞)	1.963	0.0
(1, 1)	$(12, \infty)$	2.775	(∞, ∞)	2.787	0.4
(1, 2)	$(8, \infty)$	4.336	(∞, ∞)	4.434	2.3
(1, 5)	$(5, \infty)$	8.734	$(10, \infty)$	9.092	4.1
(1, 10)	$(4, \infty)$	15.70	$(5, \infty)$	15.89	1.2

$\lambda_1 = \lambda_2 = 0.8$; $\beta_1 = 0.9$; $\beta_2 = 0.1$; $s_1 = s_2 = 0.4$; $\rho = 0.8$.

	optimum		approximation		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	(∞, ∞)	3.600	(∞, ∞)	3.600	0.0
(1, 0.2)	(∞, ∞)	4.549	(∞, ∞)	4.549	0.0
(1, 0.5)	$(17, \infty)$	6.104	$(20, \infty)$	6.145	0.7
(1, 1)	$(11, \infty)$	8.174	$(13, \infty)$	8.207	0.4
(1, 2)	$(9, \infty)$	11.58	$(9, \infty)$	11.58	0.0
(1, 5)	$(6, \infty)$	20.17	$(6, \infty)$	20.17	0.0
(1, 10)	$(5, \infty)$	32.83	$(5, \infty)$	32.83	0.0

$\lambda_1 = 0.765$; $\lambda_2 = 0.085$; $\beta_1 = \beta_2 = 0.1$;
$s_1 = s_2 = 0.1$; $\rho = 0.085$.

	optimum		approximation		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	$(\infty, 2)$	0.124	$(\infty, 1)$	0.124	0.0
(1, 0.2)	(∞, ∞)	0.125	(∞, ∞)	0.125	0.0
(1, 0.5)	(∞, ∞)	0.130	(∞, ∞)	0.130	0.0
(1, 1)	(∞, ∞)	0.137	(∞, ∞)	0.137	0.0
(1, 2)	(∞, ∞)	0.151	(∞, ∞)	0.151	0.0
(1, 5)	(∞, ∞)	0.194	(∞, ∞)	0.194	0.0
(1, 10)	(∞, ∞)	0.266	(∞, ∞)	0.266	0.0

$\lambda_1 = 0.765$; $\lambda_2 = 0.085$; $\beta_1 = \beta_2 = 0.5$;
$s_1 = s_2 = 0.1$; $\rho = 0.425$.

	optimum		approximation		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	$(\infty, 1)$	0.390	$(\infty, 1)$	0.390	0.0
(1, 0.2)	$(\infty, 1)$	0.397	$(\infty, 1)$	0.397	0.0
(1, 0.5)	$(\infty, 1)$	0.419	$(\infty, 1)$	0.419	0.0
(1, 1)	(∞, ∞)	0.453	(∞, ∞)	0.453	0.0
(1, 2)	$(10, \infty)$	0.518	(∞, ∞)	0.519	0.2
(1, 5)	$(4, \infty)$	0.684	$(7, \infty)$	0.697	1.9
(1, 10)	$(2, \infty)$	0.918	$(2, \infty)$	0.918	0.0

$\lambda_1 = 0.765$; $\lambda_2 = 0.085$; $\beta_1 = \beta_2 = 1$;
$s_1 = s_2 = 0.1$; $\rho = 0.85$.

	optimum		approximation		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	$(\infty, 1)$	3.139	$(\infty, 1)$	3.139	0.0
(1, 0.2)	$(\infty, 1)$	3.381	$(\infty, 1)$	3.381	0.0
(1, 0.5)	$(\infty, 2)$	4.069	$(\infty, 1)$	4.106	0.9
(1, 1)	(∞, ∞)	5.031	(∞, ∞)	5.031	0.0
(1, 2)	$(19, \infty)$	5.745	$(17, \infty)$	5.840	1.7
(1, 5)	$(8, \infty)$	7.023	$(7, \infty)$	7.042	0.3
(1, 10)	$(6, \infty)$	8.426	$(4, \infty)$	8.630	2.4

$\lambda_1 = 0.765$; $\lambda_2 = 0.085$; $\beta_1 = \beta_2 = 0.5$;
$s_1 = s_2 = 0.4$; $\rho = 0.425$.

	optimum		approximation		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	$(\infty, 1)$	0.753	$(\infty, 1)$	0.753	0.0
(1, 0.2)	$(\infty, 1)$	0.769	$(\infty, 1)$	0.769	0.0
(1, 0.5)	$(\infty, 4)$	0.807	(∞, ∞)	0.807	0.0
(1, 1)	(∞, ∞)	0.869	(∞, ∞)	0.869	0.0
(1, 2)	(∞, ∞)	0.994	(∞, ∞)	0.994	0.0
(1, 5)	$(10, \infty)$	1.361	(∞, ∞)	1.367	0.4
(1, 10)	$(6, \infty)$	1.933	(∞, ∞)	1.990	2.9

$\lambda_1 = 0.765$; $\lambda_2 = 0.085$; $\beta_1 = \beta_2 = 1$;
$s_1 = s_2 = 0.4$; $\rho = 0.85$.

	optimum		approximation		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	$(\infty, 2)$	3.685	$(\infty, 1)$	3.730	1.2
(1, 0.2)	$(\infty, 2)$	3.971	$(\infty, 2)$	3.971	0.0
(1, 0.5)	$(\infty, 8)$	4.698	(∞, ∞)	4.778	1.7
(1, 1)	(∞, ∞)	5.673	(∞, ∞)	5.673	0.0
(1, 2)	$(19, \infty)$	7.313	(∞, ∞)	7.464	2.0
(1, 5)	$(19, \infty)$	9.370	$(20, \infty)$	9.592	2.4
(1, 10)	$(14, \infty)$	12.59	$(14, \infty)$	12.59	0.0

$\lambda_1 = 0.765$; $\lambda_2 = 0.085$; $\beta_1 = \beta_2 = 1$;
$s_1 = 0.1$; $s_2 = 0.7$; $\rho = 0.85$.

	optimum		approximation		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	$(\infty, 2)$	3.740	$(\infty, 1)$	3.789	1.3
(1, 0.2)	$(\infty, 2)$	4.032	$(\infty, 2)$	4.032	0.0
(1, 0.5)	$(\infty, 8)$	4.776	(∞, ∞)	4.855	1.7
(1, 1)	(∞, ∞)	5.769	(∞, ∞)	5.769	0.0
(1, 2)	$(19, \infty)$	7.543	(∞, ∞)	7.597	0.7
(1, 5)	$(19, \infty)$	9.646	$(20, \infty)$	9.786	1.5
(1, 10)	$(14, \infty)$	12.86	$(14, \infty)$	12.86	0.0

$\lambda_1 = 0.5$; $\lambda_2 = 0.25$; $\beta_1 = \beta_2 = 1$;
$s_1 = 0.1$; $s_2 = 0.2$; $\rho = 0.75$.

	optimum		approximation		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	$(\infty, 1)$	1.137	$(\infty, 1)$	1.137	0.0
(1, 0.2)	$(\infty, 2)$	1.362	$(\infty, 1)$	1.389	2.0
(1, 0.5)	$(\infty, 3)$	1.895	$(\infty, 3)$	1.895	0.0
(1, 1)	(∞, ∞)	2.575	(∞, ∞)	2.575	0.0
(1, 2)	$(6, \infty)$	3.409	$(8, \infty)$	3.427	0.5
(1, 5)	$(3, \infty)$	5.035	$(3, \infty)$	5.035	0.0
(1, 10)	$(2, \infty)$	7.262	$(2, \infty)$	7.262	0.0

$\lambda_1 = 0.5$; $\lambda_2 = 1$; $\beta_1 = 1$; $\beta_2 = 0.3$;
$s_1 = 0.2$; $s_2 = 0.6$; $\rho = 0.8$.

	optimum		approximation		
(c_1, c_2)	(k_1, k_2)	cost	(k_1, k_2)	cost	%
(1, 0.1)	$(\infty, 20)$	2.188	(∞, ∞)	2.342	7.0
(1, 0.2)	$(\infty, 20)$	2.820	(∞, ∞)	2.911	3.2
(1, 0.5)	$(19, \infty)$	4.475	(∞, ∞)	4.618	3.1
(1, 1)	$(8, \infty)$	6.611	$(15, \infty)$	6.928	4.8
(1, 2)	$(6, \infty)$	9.981	$(7, \infty)$	10.11	1.3
(1, 5)	$(4, \infty)$	18.49	$(4, \infty)$	18.49	0.0
(1, 10)	$(3, \infty)$	31.66	$(3, \infty)$	31.66	0.0

TABLE 6.4. The unconstrained case; two-queue models.

$\lambda_1 = \lambda_2 = \lambda_3 = 0.25$; $\beta_1 = 0.2$; $\beta_2 = 0.6$; $\beta_3 = 2.2$;
$s_1 = s_2 = s_3 = 0.1$; $\rho = 0.75$.

	optimum		approximation		
(c_1, c_2, c_3)	(k_1, k_2, k_3)	cost	(k_1, k_2, k_3)	cost	%
(10, 10, 10)	$(\infty, \infty, 2)$	34.26	$(\infty, \infty, 2)$	34.26	0.0
(10, 3, 10)	$(\infty, 4, 2)$	29.40	$(\infty, \infty, 2)$	29.43	0.1
(10, 10, 3)	$(\infty, \infty, 1)$	18.22	$(\infty, \infty, 1)$	18.22	0.0
(10, 3, 3)	$(\infty, 8, 1)$	14.71	$(\infty, \infty, 1)$	14.72	0.0
(10, 3, 1)	$(\infty, 8, 1)$	9.28	$(\infty, \infty, 1)$	9.28	0.0
(10, 1, 3)	$(\infty, 2, 1)$	13.64	$(\infty, 3, 1)$	13.65	0.1
(10, 1, 1)	$(\infty, 3, 1)$	8.26	$(\infty, \infty, 1)$	8.28	0.3

$\lambda_1 = 0.6$; $\lambda_2 = 0.2$; $\lambda_3 = 0.05$; $\beta_1 = 0.2$; $\beta_2 = 0.6$; $\beta_3 = 2.2$;
$s_1 = s_2 = s_3 = 0.5$; $\rho = 0.35$.

	optimum		approximation		
(c_1, c_2, c_3)	(k_1, k_2, k_3)	cost	(k_1, k_2, k_3)	cost	%
(10, 10, 10)	$(\infty, \infty, 1)$	14.78	$(\infty, \infty, 1)$	14.78	0.0
(10, 3, 10)	$(\infty, 3, 1)$	12.35	$(\infty, \infty, 1)$	12.38	0.3
(10, 10, 3)	$(\infty, \infty, 1)$	13.89	$(\infty, \infty, 1)$	13.89	0.0
(10, 3, 3)	$(\infty, 3, 1)$	11.47	$(\infty, \infty, 1)$	11.49	0.2
(10, 3, 1)	$(\infty, 3, 1)$	11.22	$(\infty, \infty, 1)$	11.24	0.2
(10, 1, 3)	$(\infty, 2, 1)$	10.66	$(\infty, 1, 1)$	10.78	1.1
(10, 1, 1)	$(\infty, 2, 1)$	10.41	$(\infty, 1, 1)$	10.54	1.2

$\lambda_1 = 0.6$; $\lambda_2 = 0.2$; $\lambda_3 = 0.05$; $\beta_1 = \beta_2 = \beta_3 = 1$;
$s_1 = s_2 = s_3 = 0.1$; $\rho = 0.85$.

	optimum		approximation		
(c_1, c_2, c_3)	(k_1, k_2, k_3)	cost	(k_1, k_2, k_3)	cost	%
(10, 10, 10)	(∞, ∞, ∞)	53.05	(∞, ∞, ∞)	53.05	0.0
(10, 3, 10)	$(\infty, 2, \infty)$	32.78	$(\infty, 2, \infty)$	32.78	0.0
(10, 10, 3)	$(\infty, \infty, 1)$	45.06	$(\infty, \infty, 1)$	45.06	0.0
(10, 3, 3)	$(\infty, 3, 1)$	30.04	$(\infty, 2, 1)$	30.13	0.3
(10, 3, 1)	$(\infty, 3, 1)$	28.53	$(\infty, 2, 1)$	28.97	1.5
(10, 1, 3)	$(\infty, 1, 2)$	22.19	$(\infty, 1, 2)$	22.19	0.0
(10, 1, 1)	$(\infty, 2, 1)$	21.26	$(\infty, 1, 1)$	21.35	0.4

$\lambda_1 = 0.3$; $\lambda_2 = 0.8$; $\lambda_3 = 0.1$; $\beta_1 = 0.2$; $\beta_2 = 0.5$; $\beta_3 = 2$;
$s_1 = 2$; $s_2 = 0.1$; $s_3 = 0.5$; $\rho = 0.66$.

	optimum		approximation		
(c_1, c_2, c_3)	(k_1, k_2, k_3)	cost	(k_1, k_2, k_3)	cost	%
(10, 10, 10)	$(\infty, \infty, 2)$	63.47	$(\infty, \infty, 4)$	65.59	3.3
(10, 3, 10)	$(\infty, 18, 4)$	37.30	(∞, ∞, ∞)	37.76	1.2
(10, 10, 3)	$(\infty, \infty, 2)$	55.48	$(\infty, \infty, 2)$	55.48	0.0
(10, 3, 3)	$(\infty, \infty, 2)$	31.60	$(\infty, \infty, 2)$	31.60	0.0
(10, 3, 1)	$(\infty, \infty, 2)$	29.32	$(\infty, \infty, 2)$	29.32	0.0
(10, 1, 3)	$(\infty, 14, 2)$	24.55	$(\infty, \infty, 3)$	25.27	2.9
(10, 1, 1)	$(\infty, 16, 2)$	22.46	$(\infty, \infty, 2)$	22.50	0.2

TABLE 6.5. The unconstrained case; three-queue models; exponential service times.

$\lambda_1 = \lambda_2 = \lambda_3 = 0.25$; $\beta_1 = 0.2$; $\beta_2 = 0.6$; $\beta_3 = 2.2$;
$s_1 = s_2 = s_3 = 0.1$; $\rho = 0.75$.

	optimum		approximation		
(c_1, c_2, c_3)	(k_1, k_2, k_3)	cost	(k_1, k_2, k_3)	cost	%
(10, 10, 10)	$(\infty, \infty, 2)$	27.23	$(\infty, \infty, 2)$	27.23	0.0
(10, 3, 10)	$(\infty, 4, 2)$	23.21	$(\infty, \infty, 2)$	23.23	0.1
(10, 10, 3)	$(\infty, \infty, 1)$	14.46	$(\infty, \infty, 1)$	14.46	0.0
(10, 3, 3)	$(\infty, 8, 1)$	11.70	$(\infty, \infty, 1)$	11.70	0.0
(10, 3, 1)	$(\infty, 14, 1)$	7.440	$(\infty, \infty, 1)$	7.440	0.0
(10, 1, 3)	$(\infty, 2, 1)$	10.86	$(\infty, 3, 1)$	10.88	0.2
(10, 1, 1)	$(\infty, 3, 1)$	6.635	$(\infty, \infty, 1)$	6.652	0.3

$\lambda_1 = 0.6$; $\lambda_2 = 0.2$; $\lambda_3 = 0.05$; $\beta_1 = 0.2$; $\beta_2 = 0.6$; $\beta_3 = 2.2$;
$s_1 = s_2 = s_3 = 0.5$; $\rho = 0.35$.

	optimum		approximation		
(c_1, c_2, c_3)	(k_1, k_2, k_3)	cost	(k_1, k_2, k_3)	cost	%
(10, 10, 10)	$(\infty, \infty, 1)$	13.79	$(\infty, \infty, 1)$	13.79	0.0
(10, 3, 10)	$(\infty, 3, 1)$	11.54	$(\infty, \infty, 1)$	11.56	0.2
(10, 10, 3)	$(\infty, \infty, 1)$	12.97	$(\infty, \infty, 1)$	12.97	0.0
(10, 3, 3)	$(\infty, 3, 1)$	10.72	$(\infty, \infty, 1)$	10.74	0.1
(10, 3, 1)	$(\infty, 4, 1)$	10.49	$(\infty, \infty, 1)$	10.50	0.1
(10, 1, 3)	$(\infty, 2, 1)$	9.979	$(\infty, 1, 1)$	10.09	1.1
(10, 1, 1)	$(\infty, 2, 1)$	9.751	$(\infty, 1, 1)$	9.875	1.3

$\lambda_1 = 0.6$; $\lambda_2 = 0.2$; $\lambda_3 = 0.05$; $\beta_1 = \beta_2 = \beta_3 = 1$;
$s_1 = s_2 = s_3 = 0.1$; $\rho = 0.85$.

	optimum		approximation		
(c_1, c_2, c_3)	(k_1, k_2, k_3)	cost	(k_1, k_2, k_3)	cost	%
(10, 10, 10)	(∞, ∞, ∞)	41.03	(∞, ∞, ∞)	41.03	0.0
(10, 3, 10)	$(\infty, 2, \infty)$	25.72	$(\infty, 2, \infty)$	25.72	0.0
(10, 10, 3)	$(\infty, \infty, 1)$	35.32	$(\infty, \infty, 1)$	35.32	0.0
(10, 3, 3)	$(\infty, 2, 1)$	23.57	$(\infty, 2, 1)$	23.57	0.0
(10, 3, 1)	$(\infty, 3, 1)$	22.44	$(\infty, 2, 1)$	22.65	0.9
(10, 1, 3)	$(\infty, 1, 2)$	17.34	$(\infty, 1, 2)$	17.34	0.0
(10, 1, 1)	$(\infty, 1, 1)$	16.67	$(\infty, 1, 1)$	16.67	0.0

$\lambda_1 = 0.3$; $\lambda_2 = 0.8$; $\lambda_3 = 0.1$; $\beta_1 = 0.2$; $\beta_2 = 0.5$; $\beta_3 = 2$; $s_1 = 2$; $s_2 = 0.1$; $s_3 = 0.5$; $\rho = 0.66$.					
	optimum		approximation		
(c_1, c_2, c_3)	(k_1, k_2, k_3)	cost	(k_1, k_2, k_3)	cost	%
(10, 10, 10)	$(\infty, \infty, 3)$	55.40	$(\infty, \infty, 4)$	55.94	1.0
(10, 3, 10)	$(\infty, 18, \infty)$	33.93	(∞, ∞, ∞)	34.36	1.2
(10, 10, 3)	$(\infty, \infty, 2)$	48.62	$(\infty, \infty, 2)$	48.62	0.0
(10, 3, 3)	$(\infty, \infty, 2)$	28.69	$(\infty, \infty, 2)$	28.69	0.0
(10, 3, 1)	$(\infty, \infty, 2)$	26.67	$(\infty, \infty, 2)$	26.67	0.0
(10, 1, 3)	$(\infty, 14, 2)$	22.39	$(\infty, \infty, 3)$	23.27	3.9
(10, 1, 1)	$(\infty, 16, 2)$	20.48	$(\infty, \infty, 2)$	20.98	2.4

TABLE 6.6. The unconstrained case; three-queue models; Erlang-2 service times.

$\lambda_1 = 0.35$; $\lambda_2 = \ldots = \lambda_5 = 0.1$; $\beta_1 = 1$; $\beta_2 = \ldots = \beta_5 = 1$; $s_1 = 0.1$; $s_2 = \ldots = s_5 = 0.05$; $\rho = 0.75$.					
	optimum		approximation		
(c_1, c_{2-5})	(k_1, k_{2-5})	cost	(k_1, k_{2-5})	cost	%
(1, 0.1)	$(\infty, 1)$	0.882	$(\infty, 1)$	0.882	0.0
(1, 0.5)	$(\infty, 3)$	1.776	$(\infty, 4)$	1.776	0.0
(1, 1)	(∞, ∞)	2.623	(∞, ∞)	2.623	0.0
(1, 2)	$(3, \infty)$	3.898	$(4, \infty)$	3.898	0.0

TABLE 6.7. The unconstrained case; a five-queue model.

Discussion of the numerical results.
The proposed approach performs extremely well; in the majority of the 151 examples the achieved waiting cost is less than 1% larger than the minimal waiting cost. Only twice the achieved waiting cost is more than 5% larger than the minimal waiting cost, not once more than 10% larger.
The optimal service limits as well as the service limits obtained from the Fuhrmann & Wang approximation always satisfied the property stated in Conjecture 6.5.2, i.e., if $c_i/\beta_i = \max_{j=1,\ldots,n} c_j/\beta_j$, then $k_i^* = \infty$.
The results for an Erlang-2 service time distribution are similar to the results for an exponential service time distribution. The waiting cost for an Erlang-2 service time distribution is always smaller than the waiting cost for an exponential service time distribution with the same mean. Intuitively the waiting times are indeed likely to be smaller when the variance of the service time distribution is smaller. The optimal service limits for an Erlang-2 service time distribution however hardly differ from the optimal service limits for an exponential service time distribution with the same mean.

6.7 CONCLUDING REMARKS AND SUGGESTIONS FOR FURTHER RESEARCH

We have studied the problem of finding the optimal service limits in a cyclic polling system with the k-limited service discipline. The use of the Fuhrmann & Wang approximation is shown to be very effective in finding the optimal service limits. In the numerical experiments we have observed that the waiting cost according to the Fuhrmann & Wang approximation sometimes differs dramatically from the 'true' waiting cost obtained from the PSA, but that still the optimal service limits according to the Fuhrmann & Wang approximation agree with the 'true' optimal service limits obtained from the PSA.

Even when completely misjudging the mean waiting time, the Fuhrmann & Wang approximation apparently *does* capture the major factors important for efficient operation of the system.

The waiting-time approximation for the Bernoulli service discipline that Blanc & Van der Mei [23] use to find the optimal Bernoulli parameters q_i, and the Fuhrmann & Wang approximation that we use, coincide when $q_i = 1 - 1/k_i$, cf. Remark 6.3.1. The effectiveness of both approximations suggests that, as far as optimization is concerned, the Bernoulli service discipline is a very good emulation of the k-limited service discipline. Yet, as far as evaluation of the mean waiting time is concerned, the Bernoulli service discipline is often not a very good approximation of the k-limited service discipline. Due to the stochastic nature of the Bernoulli service discipline the mean waiting times tend to be larger than for the k-limited service discipline, cf. Tedijanto [181] Chapter 5, [182], [183].

In the present study we have been concerned with optimization of the *service discipline*, ranging from 1-limited to exhaustive, for a given cyclic server routing. Earlier studies mostly were concerned with optimization of the *server routing* for a given service discipline, like 1-limited, gated, or exhaustive, cf. Section 1.5. We feel that it would also be worthwhile to consider simultaneous optimization of the server routing and the service discipline. Simultaneous optimization of the number of visits and the amount of service per visit would enable more flexible priorization of the various stations.

At a few instances we faced difficult monotonicity questions: monotonicity of $\mathrm{E}\mathbf{W}_i$ in k_j, monotonicity of the mean waiting time for an $M/G/1$ queue with vacations in the vacation time variance. Relatively few monotonicity results for polling and vacation models have been obtained; this seems an interesting area for further research.

Chapter 7

Optimization of fixed time polling schemes

7.1 Introduction

In the present chapter we consider a polling system operated with a fixed time polling (ftp) scheme. An ftp scheme specifies which queue should be visited at what time, i.e., it specifies not only the *order* of the visits, but also the *starting times* of the visits. We are interested in the problem of constructing ftp schemes that contribute to an efficient operation of the system.
As a major benefit, ftp schemes provide by definition a guarantee on the server return time, i.e., the time until a queue receives service again. The visit times being specified beforehand, ftp schemes are also very attractive from a scheduling point of view, especially when the polling processes in question interact with other communication or production processes. Moreover, ftp schemes are very appropriate to effectuate some kind of priorization, by scheduling more visits, longer visits, or both, to high-priority queues. The consequence however is that ftp schemes may force the server to idle at an empty queue while there are customers waiting at other queues.
The similarity between ftp schemes and k-limited service, as considered in the previous chapter, lies in the limitation of the amount of service provided during a visit. The difference is that in ftp schemes the limitation refers to the visit *time* rather than the *number* of services during a visit, as in k-limited service, and that the visit is not completed until the service limit is reached, even when the queue becomes empty.
The present chapter originates from consultancy work done by order of PTT Research, Groningen. In the course of 1992 PTT Telecom started to gradually introduce itemized telephone billing in the Netherlands. To transfer the call data involved from the telephone switches to the billing center, PTT Telecom

applies a polling technique similar to the polling data link control scheme described in Section 1.1. According to a fixed time schedule, a so-called mediation system polls the telephone switches. When polled by the mediation system, the telephone switches may start to transmit files with call records to the billing center. For efficiency reasons the delays occurring in collecting the call data should be kept within bounds.

To the best of the author's knowledge, ftp schemes have not been studied before in a strict polling context. However, in other multi-class queueing settings various similar disciplines have been considered under a variety of names. Very similar to ftp schemes are e.g. so-called clocked schedules, studied in [1], [82], [96]. However, in these studies the multi-class queueing structure corresponds to a priority discipline in accordance with the timing requirements of tasks in real-time systems, rather than a distinction between separate queues like in a polling context. Therefore these studies are primarily concerned with the short-term delay of high-priority tasks with extremely stringent timing requirements, or with the probability of urgent tasks not getting served in the scheduled interval, or with the long-term delay of low-priority tasks with more liberal timing requirements, rather than with achieving a minimal overall time delay. Also related to ftp schemes are so-called Time Division Multiplexing (TDM) protocols, considered e.g. in [118], [120], [156].

It is most unlikely that ftp schemes allow an exact analysis, except for deterministic arrival, service, and switch-over processes. Note that ftp schemes do not satisfy Property 1.4.1, which provides a global characterization of the class of service disciplines that are amenable to an exact analysis. We therefore feel justified in resorting to approximations for the mean waiting times. As we are primarily concerned with minimizing a weighted sum of the mean waiting times rather than evaluating the mean waiting times themselves, we deliberately seek very simple approximations, which do not necessarily need to be very accurate, as long as they rightly capture the *behavior* of the mean waiting times.

The remainder of the chapter is organized as follows. In Section 7.2 we present a detailed model description. In Section 7.3 we formulate the problem of constructing an efficient ftp scheme as a mathematical program. In view of its NP-hardness we develop in Section 7.4 a heuristic method for solving the mathematical program. In Section 7.5 we give an overview of the numerical results that we gathered to validate the method.

7.2 Model description

The model under consideration consists of n queues, $Q_1, \ldots, Q_n$, each of infinite capacity, attended by a single server S. For the specification of the arrival, service, and switch-over processes we refer to the description of the 'basic' model in Section 1.3, but in the present chapter we do not assume that the customers arrive according to Poisson processes.

The server operates according to an ftp scheme, which specifies not only the order of the visits, but also the starting times of the visits. We represent an ftp scheme by a vector pair (P,T), $P_k \in \{1,\ldots,n\}$, $T_k \geq 0$, $k = 1,\ldots,m$. The vector P contains the polling table associated with the ftp scheme, i.e., the k-th visit is to queue P_k, $k = 1,\ldots,m$. The vector T contains the extended visit times associated with the ftp scheme, i.e., T_k is the time between the start of the k-th visit and the start of the $(k+1)$-th visit, $k = 1,\ldots,m$, where $m+1$ is to be understood as 1.

To supplement an ftp scheme, it should be specified which customers qualify for service during a visit. In addition it should be specified what is to be done in case the available visit time expires before all entitled customers have been served. In this chapter we assume that service is gated, i.e., only customers that were present at the start of a visit qualify for service during the visit. In case the available visit time expires before all entitled customers have been served, we assume that the next visit is postponed. To still fulfil the guarantee on the starting times, we demand however that the available visit times are most rarely exceeded, i.e., we demand that the available visit times are sufficiently long to meet the gated service discipline (with high probability). These assumptions were made in the light of the actual telecommunication application that motivated the present study, cf. Section 7.1.

In this chapter we are interested in the problem of finding an ftp scheme that minimizes the mean total waiting cost per unit of time. Denote by $\mathbf{W}_i$ the waiting time of an arbitrary type-i customer, $i = 1,\ldots,n$. Let c_i represent the waiting cost per unit of time of a type-i customer. The mean total waiting cost per unit of time amounts to $\sum_{i=1}^{n} c_i \lambda_i \mathrm{E}\mathbf{W}_i$.

7.3 Constructing an efficient ftp scheme I

As stated in the previous section, we are interested in the problem of constructing an ftp scheme that minimizes the mean total waiting cost per unit of time, under the side-constraint of most rarely exceeding the available visit times. Starting from rather simple approximations, we formulate in this section the problem under consideration as a mathematical program. In view of its NP-hardness we describe in the next section a heuristic method for solving the mathematical program. The approach bears resemblance to the approach in Kruskal [134] to the problem of determining efficient visit orders for polling systems with deterministic arrival, service, and switch-over processes, and the approach in Boxma, Levy, & Weststrate [48] to a similar problem for polling systems with a Poisson arrival process and general service and switch-over processes.

We first introduce some notation. Denote by $\mathbf{U}_k$ the k-th available visit time, i.e., the k-th extended visit time minus the switch-over time into queue P_k, by $\mathbf{V}_k$ the k-th required visit time, i.e., the time the server needs to do the work during the k-th visit, and by SC_k the k-th subcycle time, i.e., the time

between the start of the k-th visit and the start of the previous visit to queue P_k, $k = 1, \ldots, m$. Denote by C the cycle time, i.e., the time the server needs to pass through the ftp scheme once. By the nature of the ftp scheme, T_k, SC_k, and C are deterministic, but $\mathbf{U}_k$ and $\mathbf{V}_k$ are not (unless respectively the switch-over process and the arrival and service processes are strictly deterministic, the interarrival time in addition being a divisor of the cycle time).
Of course T_k, $\mathbf{U}_k$, $\mathbf{V}_k$, SC_k, and C are closely related. Firstly, the k-th available visit time $\mathbf{U}_k$ is what remains of the k-th extended visit time T_k, after the switch-over time into queue P_k,

$$T_k = \mathbf{U}_k + \mathbf{S}_{P_k}, \qquad k = 1, \ldots, m, \tag{7.1}$$

so that in fact $\mathbf{U}_k$ may even take a negative value, when $\mathbf{S}_{P_k}$ happens to take a value bigger than T_k. Secondly, by the nature of the gated service discipline, the k-th required visit time $\mathbf{V}_k$ equals the amount of work that arrives at queue P_k during the k-th subcycle time SC_k, $k = 1, \ldots, m$.
The k-th subcycle time is composed of the extended visit times between the start of the k-th visit and the start of the previous visit to queue P_k,

$$SC_k = \sum_{l=1}^{m} h_{kl} T_l, \qquad k = 1, \ldots, m. \tag{7.2}$$

Here the matrix $H = (h_{kl})$ is defined by

$$h_{kl} := \begin{cases} 1 & \text{if } P_{l+1}, \ldots, P_{k-1} \neq P_k \\ 0 & \text{otherwise} \end{cases} \qquad k, l = 1, \ldots, m,$$

i.e., h_{kl} indicates whether the l-th extended visit time belongs to the k-th subcycle time: $h_{kl} = 0$ iff the k-th subcycle time begins after the start of the l-th visit.
The cycle time is composed of all the extended visit times,

$$C = \sum_{l=1}^{m} T_l. \tag{7.3}$$

The cycle time may as well be viewed as consisting of all the subcycle times corresponding to any Q_i,

$$C = \sum_{\{k:P_k=i\}} SC_k, \qquad i = 1, \ldots, n. \tag{7.4}$$

Note that substituting (7.2) into (7.4) indeed yields (7.3), as $\sum_{\{k:P_k=i\}} h_{kl} = 1$, $i = 1, \ldots, n$, $l = 1, \ldots, m$.

To be able to formulate the problem as a mathematical program, we now express the mean total waiting cost per unit of time, as well as the side-constraint of most rarely exceeding the available visit times, in terms of the ftp scheme.

To start with the latter, we may represent the side-constraint by

$$\Pr\{\mathbf{U}_k < \mathbf{V}_k\} \leq \pi_{P_k}, \qquad k = 1, \ldots, m, \tag{7.5}$$

with π_i, $i = 1, \ldots, n$, prespecified bounds for the probabilities of exceeding the available visit times. As (7.5) does not really fit into the framework of a mathematical program, we replace (7.5) by a constraint of the type

$$\mathrm{E}\mathbf{U}_k - Y_k \geq \mathrm{E}\mathbf{V}_k + Z_k, \qquad k = 1, \ldots, m.$$

Here Y_k and Z_k are measures for the variability of respectively $\mathbf{U}_k$ and $\mathbf{V}_k$ such that $\Pr\{\mathbf{U}_k < \mathrm{E}\mathbf{U}_k - Y_k\} \leq \gamma_{P_k}\pi_{P_k}$ and $\Pr\{\mathbf{V}_k > \mathrm{E}\mathbf{V}_k + Z_k\} \leq (1 - \gamma_{P_k})\pi_{P_k}$, with $0 \leq \gamma_i \leq 1$, $i = 1, \ldots, n$. Thus we slightly strengthen (7.5), as $\Pr\{\mathbf{U}_k \geq \mathbf{V}_k\} \geq \Pr\{\mathbf{U}_k \geq \mathrm{E}\mathbf{U}_k - Y_k, \mathbf{V}_k \leq \mathrm{E}\mathbf{V}_k + Z_k\} = \Pr\{UU_k \geq \mathrm{E}\mathbf{U}_k - Y_k\} \Pr\{\mathbf{V}_k \leq \mathrm{E}\mathbf{V}_k + Z_k\} = 1 - \pi_{P_k} + \gamma_{P_k}(1 - \gamma_{P_k})\pi_{P_k}^2$. From (7.1) we have $\mathrm{E}\mathbf{U}_k = T_k - s_{P_k}$, $k = 1, \ldots, m$. As $\mathbf{V}_k$ equals the amount of work that arrives at Q_{P_k} during SC_k, we have $\mathrm{E}\mathbf{V}_k = \rho_{P_k} SC_k$, $k = 1, \ldots, m$. We further take $Y_k = \delta_{P_k} s_{P_k}$, $k = 1, \ldots, m$, and $Z_k = \epsilon_{P_k}\rho_{P_k} SC_k$, $k = 1, \ldots, m$. Here δ_i and ϵ_i are measures for the variability of respectively the switch-over times into Q_i and the interarrival and service times at Q_i such that

$$\Pr\{\mathbf{S}_i > (1 + \delta_i)s_i\} \leq \gamma_i\pi_i,$$

$$\sum_{h=0}^{\infty} \left(A_i^{h*}(\hat{SC}_i) - A_i^{(h+1)*}(\hat{SC}_i)\right)\left(1 - B_i^{(h+1)*}(\rho_i(1 + \epsilon_i)\hat{SC}_i)\right) \leq (1 - \gamma_i)\pi_i,$$

with $F^{j*}(\cdot)$ denoting the j-fold convolution of $F(\cdot)$. Here $\hat{SC}_i$ is a rough estimate for the subcycle times corresponding to Q_i, preferably as pessimistic as possible from the viewpoint of determining ϵ_i. Thus we slightly strengthen (7.5) further as

$$\Pr\{\mathbf{U}_k < \mathrm{E}\mathbf{U}_k - Y_k\} = \Pr\{\mathbf{S}_{P_k} > (1 + \delta_{P_k})s_{P_k}\},$$

$$\Pr\{\mathbf{V}_k > \mathrm{E}\mathbf{V}_k + Z_k\} = \Pr\{\mathbf{V}_k > \rho_{P_k}(1 + \epsilon_{P_k})SC_k\},$$

$$\Pr\{\mathbf{V}_k > t\} \leq \sum_{h=0}^{\infty} \left(A_{P_k}^{h*}(SC_k) - A_{P_k}^{(h+1)*}(SC_k)\right)\left(1 - B_{P_k}^{(h+1)*}(t)\right), \quad t \geq 0,$$

particularly for $t = \rho_{P_k}(1+\epsilon_{P_k})SC_k$. The latter inequality holds, since $A_{P_k}^{h*}(SC_k) - A_{P_k}^{(h+1)*}(SC_k)$ is the probability that $(h + 1)$ type-P_k customers arrive during SC_k, under the pessimistic assumption that the first customer arrives at the beginning of SC_k, and $1 - B_{P_k}^{(h+1)*}(t)$ is the probability that the service of $(h + 1)$ type-P_k customers is not finished within time t, $t \geq 0$.
Instead of $Z_k = \epsilon_{P_k}\rho_{P_k} SC_k$ one might take e.g. $Z_k = \epsilon_{P_k}\rho_{P_k} SC_k + (1+\zeta_{P_k})\beta_{P_k}$. Here ζ_i is a measure for the variability of the service times at Q_i such that $1 - B_i((1 + \zeta_i)\beta_i) \leq (1 - \gamma_i)\pi_i$, $i = 1, \ldots, n$. The factor $(1 + \zeta_i)\beta_i$ would avoid that $\epsilon_i \to \infty$ in the hypothetic situation that $\hat{SC}_i \downarrow 0$. If indeed one takes

$Z_k = \epsilon_{P_k}\rho_{P_k}SC_k + (1+\zeta_{P_k})\beta_{P_k}$ instead of $Z_k = \epsilon_{P_k}\rho_{P_k}SC_k$, then everywhere $(1+\delta_i)s_i$ is to be replaced by $(1+\delta_i)s_i + (1+\zeta_i)\beta_i$.
When the distributions of the interarrival, service, and switch-over times are specified, there are no serious complications in expressing δ_i and ϵ_i in terms of π_i. Nevertheless, in real life one may rather determine δ_i and ϵ_i empirically than make a questionable assumption about the distributions of the interarrival, service, and switch-over times, needed in expressing δ_i and ϵ_i in terms of π_i.
Concluding, we represent the side-constraint of most rarely exceeding the available visit times by

$$T_k \geq \rho_{P_k}(1+\epsilon_{P_k})SC_k + (1+\delta_{P_k})s_{P_k}, \qquad k = 1,\ldots,m. \tag{7.6}$$

Note that summing (7.6) with respect to $k = 1,\ldots,m$, using (7.3) and (7.4), yields $C \geq \left(\rho + \sum_{i=1}^{n}\epsilon_i\rho_i\right)C + \sum_{i=1}^{n} m_i(1+\delta_i)s_i$. Apparently $\rho + \sum_{i=1}^{n}\epsilon_i\rho_i < 1$ is a necessary condition for the extended visit times to be all non-negative. In the proof of Lemma 7.4.1 it will also appear to be a sufficient condition. In the sequel the condition $\rho + \sum_{i=1}^{n}\epsilon_i\rho_i < 1$ is always assumed to hold.
We now express the mean total waiting cost per unit of time in terms of the ftp scheme. To approximate the waiting time of an arbitrary type-i customer, we condition on the event that the type-i customer in question arrives during the k-th subcycle time SC_k with $P_k = i$. The latter event occurs with probability SC_k/C, as the probability that a customer arrives in a specific subcycle is proportional to the length of the subcycle, irrespective of the nature of the arrival process (unless the arrival process is strictly deterministic, the interarrival time in addition being a divisor of the cycle time). Further we act as if the available visit times are never exceeded. The waiting time of an arbitrary type-i customer that arrives during the k-th subcycle time SC_k with $P_k = i$, is then composed of:
i. the time from its arrival to the start of the next visit to Q_i, i.e., the residual lifetime $\mathbf{RSC}_k$ of SC_k at the arrival epoch of the customer;
ii. the time from the start of the next visit to Q_i to the start of its service, i.e., the time the server needs to do the work $\mathbf{PV}_k$ that arrived at Q_i during the past lifetime $\mathbf{PSC}_k$ of SC_k at the arrival epoch of the customer.
As SC_k is deterministic, $\mathrm{E}\mathbf{RSC}_k = \frac{1}{2}SC_k$, $\mathrm{E}\mathbf{PSC}_k = \frac{1}{2}SC_k$. Further we use the approximation $\mathrm{E}\mathbf{PV}_k \approx \rho_i \mathrm{E}\mathbf{PSC}_k$, which in fact is exact for a Poisson arrival process, cf. Boxma et al. [48]. For less variable arrival processes the approximation will probably result in a slight overestimation, which however will tend to zero for SC_k sufficiently large compared to $1/\lambda_i$, cf. Kruskal [134].
Concluding, we approximate the mean waiting time of an arbitrary type-i customer by

$$\mathrm{E}\mathbf{W}_i \approx \frac{1+\rho_i}{2C}\sum_{\{k:P_k=i\}} SC_k^2, \qquad i = 1,\ldots,n, \tag{7.7}$$

which yields for the mean total waiting cost per unit of time

$$\sum_{i=1}^{n} c_i \lambda_i \mathrm{E}\mathbf{W}_i \approx \frac{1}{2C} \sum_{i=1}^{n} c_i \lambda_i (1+\rho_i) \sum_{\{k:P_k=i\}} SC_k^2 \tag{7.8}$$

$$= \frac{1}{2C} \sum_{k=1}^{m} c_{P_k} \lambda_{P_k} (1+\rho_{P_k}) SC_k^2.$$

Having expressed the mean total waiting cost per unit of time, as well as the side-constraint of most rarely exceeding the available visit times, in terms of the ftp scheme, we are now able to formulate the problem as a mathematical program.

Problem (I).

$$\min \quad \frac{1}{2C} \sum_{i=1}^{n} c_i \lambda_i (1+\rho_i) \sum_{\{k:P_k=i\}} SC_k^2 = \tag{7.9}$$

$$\frac{1}{2C} \sum_{k=1}^{m} c_{P_k} \lambda_{P_k} (1+\rho_{P_k}) SC_k^2$$

$$\text{sub} \quad T_k \geq \rho_{P_k}(1+\epsilon_{P_k}) SC_k + (1+\delta_{P_k}) s_{P_k}, \qquad k=1,\ldots,m; \tag{7.10}$$

$$SC_k = \sum_{l=1}^{m} h_{kl} T_l, \qquad k=1,\ldots,m; \tag{7.11}$$

$$C = \sum_{l=1}^{m} T_l; \tag{7.12}$$

$$m_i = |\{k : P_k = i\}| \geq 1, \qquad i=1,\ldots,n; \tag{7.13}$$

$$h_{kl} = \begin{cases} 1 & \text{if } P_{l+1},\ldots,P_{k-1} \neq P_k \\ 0 & \text{otherwise} \end{cases} \qquad k,l=1,\ldots,m; \tag{7.14}$$

$$P_k \in \{1,\ldots,n\}, \qquad k=1,\ldots,m; \tag{7.15}$$

$$T_k \geq 0, \qquad k=1,\ldots,m. \tag{7.16}$$

Note that the determination of m, the length of the polling table, is part of the optimization problem.
For a specific parameter choice problem (I) amounts to the problem of partitioning $m-2$ numbers into 2 sets, such that the sums of the numbers in both sets are as equal as possible, which is known to be NP-hard; cf. Lemma 7.3.1.

Lemma 7.3.1
Problem (I) is NP-hard.

Proof
See Appendix 7.A.

□

Lemma 7.3.1 suggests that there is little hope of solving problem (I) exactly in a reasonable amount of time. In the next section we therefore describe a method for solving problem (I) approximately.

7.4 Constructing an efficient FTP scheme II

In the previous section we formulated the problem under consideration as a mathematical program. In view of its NP-hardness we describe in this section a heuristic method for solving the mathematical program. The idea is to divide problem (I) into three subproblems, which are somewhat easier to handle, viz.:

1. Determination of the visit numbers $m_1, \ldots, m_n$.
2. Determination of the visit order.
3. Determination of the extended visit times $T_1, \ldots, T_m$.

Ad 1. Determination of the visit numbers.
To simplify the determination of the visit numbers, we forget about the visit order for now. So we ignore of which extended visit times the k-th subcycle time is composed, but of course we do not ignore that the subcycle times corresponding to any Q_i together make up the cycle time, cf. (7.4). Translated to problem (I), we replace the constraints (7.11), (7.14), and (7.15) by the constraint (7.4). It is easily verified that in the resulting problem the optimal subcycle times corresponding to Q_i are all equal, $i = 1, \ldots, n$, i.e., $SC_k = C/m_{P_k}$, $k = 1, \ldots, m$. Observe that $\sum_{h=1}^{H} x_h^2$, under the constraint $\sum_{h=1}^{H} x_h = X$, is minimal for $x_h = X/H$, $h = 1, \ldots, H$. Note that $SC_k = C/m_{P_k}$, $k = 1, \ldots, m$, suggests spacing the visits to the various queues as evenly as possible, as intuitively is indeed expected to be optimal, cf. Kruskal [134], Boxma et al. [48]. As seen from (7.10) and (7.12), all the optimal extended visit times at Q_i are then equal too, $i = 1, \ldots, n$, i.e., $T_k = \rho_{P_k}(1 + \epsilon_{P_k})C/m_{P_k} + (1 + \delta_{P_k})s_{P_k}$, $k = 1, \ldots, m$. Denote by D_i the common value of all these optimal extended visit times at Q_i, $i = 1, \ldots, n$. As we forget about the visit order for now, the ultimate extended visit times at Q_i will probably deviate from D_i. Remember that because of the constraint (7.10) the extended visit times can not be determined before the visit order is determined. Nevertheless, D_i will probably be a good indication for the ultimate extended visit times at Q_i, which will be useful in determining a good visit order later on. To simplify the determination of the visit numbers even further, we relax the integrality constraint (7.13) for now too. Concluding, we formulate the problem of determining the visit numbers as follows.

Problem (II).

$$\min \quad \sum_{i=1}^{n} \frac{c_i \lambda_i (1 + \rho_i) C}{2 m_i} \tag{7.17}$$

$$\text{sub} \qquad D_i = \frac{\rho_i(1+\epsilon_i)C}{m_i} + (1+\delta_i)s_i, \qquad i = 1,\ldots,n; \tag{7.18}$$

$$C = \sum_{i=1}^{n} m_i D_i; \tag{7.19}$$

$$m_i > 0, \qquad i = 1,\ldots,n. \tag{7.20}$$

Note that the objective function as well as the constraints are homogeneous with regard to $(m_1,\ldots,m_n,C)$, as is to be expected, since concatenating an ftp scheme several times does not make any difference. So we know beforehand that the optimal solution contains a positive scaling factor with regard to $(m_1,\ldots,m_n,C)$. Using the Lagrangean multiplier technique, we find that the optimal solution is

$$D_i^* = \frac{\rho_i(1+\epsilon_i)C^*}{m_i^*} + (1+\delta_i)s_i, \qquad i = 1,\ldots,n; \tag{7.21}$$

$$C^* = \frac{R\sum_{i=1}^{n}\sqrt{c_i\lambda_i(1+\rho_i)(1+\delta_i)s_i}}{1-\rho-\sum_{i=1}^{n}\epsilon_i\rho_i}; \tag{7.22}$$

$$m_i^* = R\sqrt{\frac{c_i\lambda_i(1+\rho_i)}{(1+\delta_i)s_i}}, \qquad i = 1,\ldots,n. \tag{7.23}$$

Here R is the positive scaling factor mentioned above, due to which some freedom remains in determining the total number of visits m. One may e.g. choose R such that $|\frac{m_i^* - [m_i^*]}{m_i^*}| \leq \eta_i$, $i = 1,\ldots,n$, with $[x]$ denoting the nearest integer to x and η_i prespecified bounds for the relative deviation of the true, integer visit numbers from the desirable, generally non-integer visit numbers. Alternatively, one may choose R such that the cycle time has some desirable value, which matches e.g. a daily or hourly pattern in a specific application. For (7.21), (7.22), and (7.23), the objective function (7.17) takes the value

$$\frac{\left(\sum_{i=1}^{n}\sqrt{c_i\lambda_i(1+\rho_i)(1+\delta_i)s_i}\right)^2}{2\left(1-\rho-\sum_{i=1}^{n}\epsilon_i\rho_i\right)}. \tag{7.24}$$

As seen from the argumentation preceding the formulation of problem (II), formula (7.24) provides a lower bound for the value of (7.9) for the optimal solution of problem (I). As far as the determination of the visit numbers is concerned, problem (II) may thus be conceived as a relaxation of problem (I).

Remark 7.4.1
Apart from the slack coefficient δ_i, formula (7.23) agrees with results in Kruskal [134] for polling systems with deterministic arrival, service, and switch-over

processes, and in Boxma, Levy, & Weststrate [48] for polling systems with a Poisson arrival process and general service and switch-over processes. Kruskal [134] as well as Boxma et al. [48] are mainly interested in determining the optimal visit *numbers*, just like we are here. In fact, the determination of the visit *times* does not play an essential role in [134] and [48], as they are completely governed by the service discipline that is used (either exhaustive or gated). Here however the determination of the visit times adds an extra dimension to the problem, as they should be fixed, but still should be sufficiently long (with high probability) to not interfere with the gated service discipline, leading to the side-constraint $T_k \geq \rho_{P_k}(1+\epsilon_{P_k})SC_k + (1+\delta_{P_k})s_{P_k}$. Simply taking $T_k = \rho_{P_k}SC_k + s_{P_k}$, we find ourself in the setting of Kruskal's paper [134], where the side-constraint is automatically fulfilled by the lack of any statistical fluctuation in the arrival, service, and switch-over processes. The fact that, in spite of the differences, formula (7.23) agrees with the results in [134] and [48] suggests that the optimal visit numbers are quite robust with regard to the actual length of the visit times as well as the statistical properties of the arrival, service, and switch-over processes, which has already been argued in the study of Boxma et al. [48].

□

In the special case that we confine ourself beforehand to strictly cyclic polling, i.e., $m_i = 1$, $i = 1, \ldots, n$, problem (II) reduces to

$$\min \quad \sum_{i=1}^{n} \frac{c_i\lambda_i(1+\rho_i)C}{2} \tag{7.25}$$

$$\text{sub} \quad D_i = \rho_i(1+\epsilon_i)C + (1+\delta_i)s_i, \qquad i = 1, \ldots, n; \tag{7.26}$$

$$C = \sum_{i=1}^{n} D_i. \tag{7.27}$$

The only feasible and hence optimal solution is

$$D_i^* = \rho_i(1+\epsilon_i)C^* + (1+\delta_i)s_i, \qquad i = 1, \ldots, n; \tag{7.28}$$

$$C^* = \frac{\sum_{i=1}^{n}(1+\delta_i)s_i}{1-\rho-\sum_{i=1}^{n}\epsilon_i\rho_i}. \tag{7.29}$$

For (7.28) and (7.29), the objective function (7.25) takes the value

$$\frac{\left(\sum_{i=1}^{n} c_i\lambda_i(1+\rho_i)\right)\left(\sum_{i=1}^{n}(1+\delta_i)s_i\right)}{2\left(1-\rho-\sum_{i=1}^{n}\epsilon_i\rho_i\right)}. \tag{7.30}$$

Because of the Hölder inequality (7.30) can not be smaller than (7.24), as is to be expected, when we confine ourself to strictly cyclic polling. In fact the difference between (7.30) and (7.24) gives a rough estimate of the increase in the mean total waiting cost per unit of time, when we confine ourself to strictly cyclic polling.

Ad 2. Determination of the visit order.
To facilitate the determination of the visit order, we assume that the extended visit times at Q_i are all equal to D_i^*, the indication for the extended visit times at Q_i obtained in (7.21). Translated to problem (I), we replace the constraint (7.10) by the constraint $T_k = D_{P_k}^*$. As seen from the proof of Lemma 7.3.1 however, the determination of the optimal visit order for fixed m_i and fixed $T_k = D_{P_k}^*$ is still NP-hard. Nevertheless, we rather solve problem (I) for fixed m_i and fixed $T_k = D_{P_k}^*$ approximately than an even further garbled version of problem (I) exactly.
In Appendix 7.B we describe the Golden Ratio procedure, which is an approved method for spacing the visits to the various queues as evenly as possible, cf. [118], [120], and [156]. To be specific, define $X_k := \sum_{l=1}^{m} h_{kl}$, i.e., X_k is the number of visits between the start of the k-th visit and the start of the previous visit to queue P_k, $k = 1, \ldots, m$. The Golden Ratio procedure aims at making the numbers X_k with $P_k = i$ as equal as possible. In fact the numbers X_k with $P_k = i$ are guaranteed to take at most three different values. However, these three different values are *not* guaranteed to be all nearly equal. Moreover, the Golden Ratio procedure aims at making the *numbers* of visits X_k with $P_k = i$ as equal as possible, instead of the *periods* between visits SC_k with $P_k = i$, as we should, cf. the argumentation preceding the formulation of problem (II). In other words, the Golden Ratio procedure aims at solving problem (I) for $T_k = 1$, $k = 1, \ldots, m$, instead of $T_k = D_{P_k}^*$, $k = 1, \ldots, m$. Lastly, the Golden Ratio procedure does not take into account the coefficients $c_i\lambda_i(1 + \rho_i)$ in the objective function (7.9) to weigh the improvement in the spacing of the visits to one queue against the deterioration in the spacing of the visits to another queue. In Appendix 7.C we describe a procedure based on extremal splittings, which to some extent meets these objections. Whichever of these procedures is used, it is always worthwhile to make sure that the visit order is optimal with respect to some neighborhood. One may e.g. attempt to improve the visit order by interchanging pairs of consecutive visits.

Ad 3. Determination of the extended visit times.
At first sight it does not seem to make sense to protract a visit any longer than needed to satisfy the side-constraint of most rarely exceeding the available visit times. Remember that the server will most rarely be busy during the extra time. Still for extreme parameter choices it may make sense to protract a visit. Take e.g. $n = 101$, $\beta_i = 0$, $\delta_i = 0$, $i = 1, \ldots, n$. The side-constraint (7.5) then reduces to $T_k \geq s_{P_k}$, $k = 1, \ldots, m$. Take $c_1\lambda_1 = 10000$, $c_i\lambda_i = 1$,

$i = 2, \ldots, 100$, $c_{101}\lambda_{101} = 100$, $s_1 = 1$, $s_i = 1$, $i = 2, \ldots, 100$, $s_{101} = 100$. From (7.23) we then obtain $m_1 = 100R$, $m_i = R$, $i = 2, \ldots, 100$, $m_{101} = R$, which for $R = 1$ yields the polling table $P_{2i-1} = 1$, $i = 1, \ldots, 100$, $P_{2i-2} = i$, $i = 2, \ldots, 100$, $P_{200} = 101$. It is easily verified that (7.9) is larger for $T_k = s_{P_k}$, $k = 1, \ldots, 200$, than for $T_{2i-2} = 2 > s_{P_{2i-2}}$ (protracted visit), $T_k = s_{P_k}$, $k \neq 2i - 2$ for any $i = 2, \ldots, 100$. Nevertheless, for realistic parameter choices it will seldom really pay off to protract a visit. To facilitate the determination of the extended visit times, we therefore assume that the side-constraint of most rarely exceeding the available visit times is satisfied without any slack. Translated to problem (I), we assume that the constraint (7.10) is satisfied without any slack. Thus determining the extended visit times amounts to solving a set of linear equations; cf. Lemma 7.4.1.

Lemma 7.4.1
The set of linear equations

$$T_k = \rho_{P_k}(1 + \epsilon_{P_k}) \sum_{l=1}^{m} h_{kl} T_l + (1 + \delta_{P_k}) s_{P_k}, \qquad k = 1, \ldots, m, \tag{7.31}$$

has a unique solution; this solution is non-negative.

Proof
See Appendix 7.D.

□

Remark 7.4.2
Note that summing (7.31) with respect to $\{k : P_k = i\}$ yields for the mean total available visit time at Q_i during a cycle $\sum_{\{k:P_k=i\}} T_k - m_i s_i = \rho_i(1+\epsilon_i)C + m_i\delta_i s_i$, as $\sum_{\{k:P_k=i\}} h_{kl} = 1$, $i = 1, \ldots, n$, $l = 1, \ldots, m$. The mean total available visit time may be viewed as consisting of (i) $\rho_i C$, the time needed to satisfy the stability condition and (ii) $\epsilon_i \rho_i C + m_i \delta_i s_i$, the extra time above $\rho_i C$ needed to satisfy the side-constraint of most rarely exceeding the available visit times.

□

We finally summarize the method for constructing an ftp scheme as follows.

1. Determination of the visit numbers.
Calculate the desirable visit frequencies

$$f_i^* = \frac{\sqrt{\dfrac{c_i \lambda_i (1 + \rho_i)}{(1 + \delta_i) s_i}}}{\sum_{j=1}^{n} \sqrt{\dfrac{c_j \lambda_j (1 + \rho_j)}{(1 + \delta_j) s_j}}}, \qquad i = 1, \ldots, n. \tag{7.32}$$

Determine the total number of visits m^*.
One may choose m^* with $m^* = \sum_{i=1}^{n} [m^* f_i^*]$, $[m^* f_i^*] \geq 1$, e.g. such that

$$\frac{|f_i^* - \frac{[m^* f_i^*]}{m^*}|}{f_i^*} \leq \eta_i, \qquad i = 1, \ldots, n, \tag{7.33}$$

with $[x]$ denoting the nearest integer to x and η_i prespecified bounds for the relative deviation of the true visit frequencies from the desirable visit frequencies.
Alternatively, one may choose m^* such that

$$C^* = \frac{m^* \sum_{i=1}^{n} \sqrt{c_i \lambda_i (1+\rho_i)(1+\delta_i) s_i}}{\left(1 - \rho - \sum_{i=1}^{n} \epsilon_i \rho_i\right) \sum_{i=1}^{n} \sqrt{\frac{c_i \lambda_i (1+\rho_i)}{(1+\delta_i) s_i}}} \approx C_0,$$

with C_0 some desirable value for the cycle time.
Take $m_i^* = [m^* f_i^*]$, $i = 1, \ldots, n$.

2. Determination of the visit order.
Construct a polling table P^*, using e.g. one of the methods described in the appendices.
Calculate

$$D_i^* = \rho_i (1+\epsilon_i) \sqrt{\frac{(1+\delta_i) s_i}{c_i \lambda_i (1+\rho_i)}} \frac{\sum_{j=1}^{n} \sqrt{c_j \lambda_j (1+\rho_j)(1+\delta_j) s_j}}{1 - \rho - \sum_{j=1}^{n} \epsilon_j \rho_j} + (1+\delta_i) s_i,$$

as indication for the extended visit times at Q_i.

3. Determination of the extended visit times.
Solve the set of linear equations

$$T_k^* = \rho_{P_k^*} (1 + \epsilon_{P_k^*}) \sum_{l=1}^{m^*} h_{kl}^* T_l^* + (1 + \delta_{P_k^*}) s_{P_k^*}, \qquad k = 1, \ldots, m^*. \tag{7.34}$$

7.5 Numerical results

In the previous section we developed a method for constructing an efficient ftp scheme. In this section we give an overview of the numerical results that we gathered to validate the method. The numerical results were obtained from a simulation tool developed at PTT Research, Groningen, cf. Harink, Cramer, & Huitema [117]. Due to the complexity of the problem it seems not possible to identify the true optimal ftp scheme, neither in an analytic manner nor by

an exhaustive search. Instead we therefore compare the proposed ftp scheme with some other natural, but less sophisticated, ftp schemes. In addition we compare the proposed ftp scheme with some 'neighboring' ftp schemes, partly to test the method, partly as a possible first step in improving the method itself.

Throughout the section the service times and the switch-over times are assumed to be constant, but possibly varying from queue to queue. The interarrival times are assumed to be almost constant, but also possibly varying from queue to queue; at Q_i the interarrival times are distributed as $1/\lambda_i + \mathbf{N}_i$, with $\mathbf{N}_i$ normally distributed with mean 0 and standard deviation $1/(10\lambda_i)$. These assumptions were made in the light of the actual telecommunication application that motivated the present study, cf. Section 7.1. The cost coefficients are assumed to be $c_i = 1/\lambda$, i.e., the goal is minimizing the overall mean waiting time. For the reason mentioned above formula (7.6) we replace the the side-constraint (7.6) by $T_k \geq \rho_{P_k}(1+\epsilon_{P_k})SC_k + (1+\delta_{P_k})s_{P_k} + \beta_{P_k}$. Consequently, in comparison with Sections 7.3 and 7.4, s_i is everywhere replaced by $s_i + \beta_i$. We consider the following eight models.

I. $n = 4$; $\lambda_i = 0.75$; $\beta_i = 0.25$; $\rho_i = 0.1875$; $s_i = 0.25$; $i = 1, \ldots, n$.

II. $n = 4$; $\lambda_i = 0.375$; $\beta_i = 0.5$; $\rho_i = 0.1875$; $s_i = 0.25$; $i = 1, \ldots, n$.

III. $n = 2$; $\lambda_1 = \lambda_2 = 0.75$; $\beta_1 = \beta_2 = 0.5$; $\rho_1 = \rho_2 = 0.375$; $s_1 = 0.05$; $s_2 = 0.45$.

IV. $n = 2$; $\lambda_1 = \lambda_2 = 0.75$; $\beta_1 = 0.1$; $\beta_2 = 0.9$; $\rho_1 = 0.075$; $\rho_2 = 0.675$; $s_1 = s_2 = 0.25$.

V. $n = 2$; $\lambda_1 = 0.15$; $\lambda_2 = 1.35$; $\beta_1 = \beta_2 = 0.5$; $\rho_1 = 0.075$; $\rho_2 = 0.675$; $s_1 = s_2 = 0.25$.

VI. $n = 4$; $\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = 0.375$; $\beta_1 = \beta_2 = 0.1$; $\beta_3 = \beta_4 = 0.9$; $\rho_1 = \rho_2 = 0.0375$; $\rho_3 = \rho_4 = 0.3375$; $s_1 = s_3 = 0.05$; $s_2 = s_4 = 0.45$.

VII. $n = 4$; $\lambda_1 = \lambda_2 = 0.075$; $\lambda_3 = \lambda_4 = 0.675$; $\beta_1 = \beta_2 = \beta_3 = \beta_4 = 0.5$; $\rho_1 = \rho_2 = 0.0375$; $\rho_3 = \rho_4 = 0.3375$; $s_1 = s_3 = 0.05$; $s_2 = s_4 = 0.45$.

VIII. $n = 4$; $\lambda_1 = \lambda_2 = 0.075$; $\lambda_3 = \lambda_4 = 0.675$; $\beta_1 = \beta_3 = 0.1$; $\beta_2 = \beta_4 = 0.9$; $\rho_1 = 0.0075$; $\rho_2 = \rho_3 = 0.0675$; $\rho_4 = 0.6075$; $s_1 = s_2 = s_3 = s_4 = 0.25$.

Note that for each model $\rho = 0.75$. Further we have taken for each model $\delta_i = 0$, $\eta_i = 0.05$, $\epsilon_i = 0.01$, $i = 1, \ldots, n$.

The first two models are completely symmetric, so that the optimal ftp scheme should be of the form $m_i = 1$, $i = 1, \ldots, n$, $T_k = T$, $k = 1, \ldots, m$, for some unknown constant T. Table 7.1 contains the mean waiting time for $T = 2.062$ in model I and for $T = 3.092$ in model II, as obtained from the developed method, as well as for various larger values of T. Smaller values of T can not be chosen in view of the side-constraint $T_k \geq \rho_{P_k}(1+\epsilon_{P_k})SC_k + (1+\delta_{P_k})s_{P_k} + \beta_{P_k}$. The 95% confidence interval for the mean waiting time approximately equals the value listed in the table +/- 0.5%. Note that the mean waiting time is indeed minimal for $T = 2.062$ in model I and for $T = 3.092$ in model II.

	I
$T = 2.062$	4.777
$T = 2.100$	4.866
$T = 2.400$	5.577
$T = 2.700$	6.294
$T = 3.000$	7.003

	II
$T = 3.092$	7.111
$T = 3.200$	7.362
$T = 3.500$	8.070
$T = 4.000$	9.262
$T = 4.500$	10.45

TABLE 7.1. The mean waiting times for models I and II.

The remaining six models are asymmetric, so that the influence of the various system parameters on the optimal ftp scheme can be investigated. By means of the models III, IV, and V, we can examine the effect of variations in the arrival rate, the service times, and the switch-over times. On the basis of the models VI, VII, and VIII we can investigate the effect of pairwise variations.
For these six models, we consider the following ftp schemes. The first ftp scheme (FTP 1) is the original ftp scheme constructed by the method described in Section 7.4. In the second ftp scheme (FTP 2) the visit numbers are equal for all queues, i.e., $m_i = 1$, $i = 1, \ldots, n$ (strictly cyclic polling, which is treated as a special case in Section 7.4). The visit times are determined according to the third step of the method described in Section 7.4. In the third ftp scheme (FTP 3) the available visit times are equal for all visits, i.e., $U_k = U$, $k = 1, \ldots, m$, while the visit numbers are equal to $m_i = M\rho_i(1 + \epsilon_i)$. The extended visit times may however still differ due to the switch-dependent switch-over times: $T_k = U + s_{P_k}$. The constant U should be chosen such that $D_i \geq \rho_i(1 + \epsilon_i)C/m_i + s_i + \beta_i$. Some straightforward calculations show that this implies $U \geq \dfrac{\sum_{j=1}^{n} \rho_j(1 + \epsilon_j)s_j + \beta_i}{1 - \rho - \sum_{j=1}^{n} \epsilon_j\rho_j}$. The constant U is chosen equal to the maximum over $i = 1, \ldots, n$ of the right-hand side of this expression. The visit order is determined according to the second step of the method described in Section 7.4. The fourth ftp scheme (FTP 4) is similar to FTP 1, except that the polling table is constructed by the procedure based on extremal splittings instead of the Golden Ratio procedure.
In addition we consider some 'neighboring' ftp schemes of FTP 1, in which the number of visits to one of the queues is either incremented or decremented by 1; in the ftp scheme indicated by FTP 'mi' the number of visits to Q_i is decremented by 1; in the ftp scheme denoted by FTP 'pi' the number of visits to Q_i is incremented by 1.
To be able to judge the numerical results correctly, we first compare in Table 7.2 the desired visit frequencies f_i for FTP 1, cf. (7.32), with the realized visit frequencies $r_i = m_i/m$ for the models III through VIII.

	III	IV	V
f_1	0.5679	0.5922	0.2108
f_2	0.4321	0.4078	0.7892

	III	IV	V
r_1	0.5625	0.5938	0.2188
r_2	0.4375	0.4063	0.7813

	VI	VII	VIII
f_1	0.4252	0.1289	0.1482
f_2	0.2221	0.0981	0.0842
f_3	0.1918	0.4390	0.4578
f_4	0.1609	0.3340	0.3098

	VI	VII	VIII
r_1	0.4375	0.1250	0.1515
r_2	0.2188	0.0938	0.0808
r_3	0.1875	0.4375	0.4545
r_4	0.1563	0.3438	0.3131

TABLE 7.2. The desired and realized visit frequencies for models III-VIII.

Table 7.3 contains the mean overall waiting time for the models III through VIII. The 95% confidence interval for the overall mean waiting time approximately equals the value listed in the table +/- 0.5%. The lines indicated by (7.9) 1 and (7.9) 4 in Table 7.3 give the value of the mean waiting time approximation (7.9) for FTP 1 and FTP 4, respectively. Comparison of these lines with the preceding lines suggests that the approximation (7.9) tends to yield a slight structural overestimation of the mean waiting time, as indeed expected for almost constant interarrival times (in view of the assumption $\mathbf{EPV}_k \approx \rho_i \mathbf{EPSC}_k$ in Section 7.3). The first line in Table 7.3 gives the value of (7.24) for the various models. As remarked before, (7.24) provides a lower bound for the waiting-time approximation (7.9). In other words, the difference between the first line and the lines indicated by (7.9) 1 and (7.9) 4 provides an upper bound on how far FTP 1 and FTP 4 can be off in minimizing the waiting-time *approximation*. The difference also gives an indication how far FTP 1 and FTP 4 can at most be off in minimizing the *true* waiting time.

	III	IV	V
(7.24)	4.1756	4.1290	3.7425
FTP 1	4.1604	4.3018	3.6521
(7.9) 1	4.3997	4.5379	3.8912
FTP 2	**4.0140**	**4.0301**	4.7513
FTP 3	4.2301	19.152	4.7315
FTP 4	4.0719	4.2988	**3.5831**
(7.9) 4	4.3101	4.5366	3.8221
FTP p1	4.1863	4.3428	3.6295
FTP m1	4.0435	4.2536	3.6778
FTP p2	4.0405	4.4105	3.6490
FTP m2	4.2148	4.3880	3.6566

	VI	VII	VIII
(7.24)	6.7580	6.1170	5.9773
FTP 1	7.0327	6.2255	6.4314
(7.9) 1	7.2695	6.4654	6.6707
FTP 2	7.1341	7.8523	7.8761
FTP 3	41.244	9.1655	32.874
FTP 4	7.2130	**6.0879**	**6.2917**
(7.9) 4	7.4495	6.3243	6.5301
FTP p1	7.0837	6.2007	6.4459
FTP m1	**6.9831**	6.2815	6.4414
FTP p2	7.0140	6.3226	6.4499
FTP m2	7.1309	6.4903	6.5360
FTP p3	7.1098	6.2950	6.4560
FTP m3	7.0469	6.1906	6.4529
FTP p4	7.1538	6.1994	6.3943
FTP m4	7.4608	6.2802	6.4893

TABLE 7.3. The mean waiting times for models III-VIII.

The number printed in boldface represents the optimum. For all models FTP 1, FTP 4, and the neighboring schemes of FTP 1 give very similar results; FTP 4 tends to perform slightly better than FTP 1. Apparently the procedure based on extremal splittings indeed tends to yield a slightly better polling table than the Golden Ratio procedure. FTP 2 is on the average slightly worse, while FTP 3 is generally bad. The low variability of the arrival, service, and switch-over processes appears to result in a relative insensitivity to the right choice of the visit numbers in the neighborhood of the desired visit numbers. Remember that a wrong choice for the visit numbers may still be compensated for in the determination of the visit times.

The fact that in the models III and IV FTP 2 performs slightly better than FTP 1 and FTP 4, may be explained as follows. As observed in Section 7.4, the visits to the various queues should be spaced as evenly as possible. For FTP 2 the very nature of cyclic polling allows the visits to the various queues to be perfectly evenly spaced, whereas for FTP 1 and FTP 4 the desired visit frequencies in the models III and IV, cf. Table 7.2, do not even allow the visits to be reasonably evenly spaced. In the derivation of the desired visit frequencies the visits to the queues were however assumed to be perfectly evenly spaced. The fact that nevertheless FTP 2 performs only slightly better than FTP 1 and FTP 4, actually supports the approach used. As Table 7.3 confirms, FTP 2 is likely to outperform FTP 1 and FTP 4 only when the number of queues is small and the difference in the desired visit numbers not too large.

APPENDICES

7.A PROOF OF LEMMA 7.3.1

Lemma 7.3.1
Problem (I) is NP-hard.

Proof
To prove that problem (I) is NP-hard, we need to show that the decision variant of problem (I) is NP-complete. A decision problem is said to be NP-complete if (i) it belongs to the class NP and (ii) every problem in the class NP is (polynomially) reducible to it, cf. Garey & Johnson [105]. For brevity let us refer to the decision variant of problem (I) as the decision problem TABLE. TABLE reads as follows: given parameters λ_i, β_i, s_i, c_i, δ_i, ϵ_i, $i = 1, \ldots, n$, and an arbitrary number r, does problem (I) have a solution which is feasible and for which the value of the objective function is not larger than r?
Obviously TABLE belongs to the class NP. As the notion of reducibility is transitive, it in fact remains to be shown that *some* known NP-hard problem is (polynomially) reducible to TABLE. Here the problem PARTITION turns out to be an appropriate choice as known NP-hard problem. PARTITION reads as follows: given a set $A = \{a_1, \ldots, a_p\}$ of p integers, does A include a subset B, such that $\sum_{a_i \in B} a_i = \sum_{a_i \in A \setminus B} a_i = \frac{1}{2} \sum_{i=1}^{p} a_i$?
We now prove that PARTITION is (polynomially) reducible to TABLE. Given an instance $a_1, \ldots, a_p$ for PARTITION, construct an instance λ_i, β_i, s_i, c_i, δ_i, ϵ_i, $i = 1, \ldots, n$, and r for TABLE in the following manner.

$$
\begin{aligned}
& n := p + 1; \\
& \beta_i := 0, \delta_i := 0, \epsilon_i := 0, \qquad i = 1, \ldots, p+1; \\
& s_i := a_i, c_i \lambda_i := a_i, \qquad i = 1, \ldots, p; \\
& s_{p+1} := 1, c_{p+1} \lambda_{p+1} := 4; \\
& r := \frac{1}{2} \left(\sum_{i=1}^{p} a_i + 2 \right)^2 .
\end{aligned}
\tag{7.35}
$$

We now need to prove that $a_1, \ldots, a_p$ constitute a 'yes' instance for PARTITION iff λ_i, β_i, s_i, c_i, δ_i, ϵ_i, $i = 1, \ldots, n$, and r as defined in (7.35), constitute a 'yes' instance for TABLE. We first show that $a_1, \ldots, a_p$ constitute a 'yes' instance for PARTITION iff there exists a feasible polling scheme (P, T), such that $m_i = 1$, $i = 1, \ldots, p$, $m_{p+1} = 2$, $SC_k = \frac{1}{2} \sum_{i=1}^{p} a_i + 1$ for both k with $P_k = p + 1$.
$\{\Rightarrow\}$ The set $A = \{a_1, \ldots, a_p\}$ includes a subset B, such that $\sum_{a_i \in B} a_i =$

$\sum_{a_i \in A \setminus B} a_i = \frac{1}{2} \sum_{i=1}^{p} a_i$. Let us say $B = \{a_{i_1}, \ldots, a_{i_q}\}$, $A \setminus B = \{a_{i_{q+1}}, \ldots, a_{i_p}\}$. Take $P_1 = p+1$, $P_{k+1} = i_k$ for $k = 1, \ldots, q$, $P_{q+2} = p+1$, $P_{k+2} = i_k$ for $k = q+1, \ldots, p$, $T_k = s_{P_k}$, $k = 1, \ldots, p+2$. Then $SC_1 = \sum_{k=q+2}^{p+2} T_k = \sum_{a_i \in A \setminus B} a_i + 1 = \frac{1}{2} \sum_{i=1}^{p} a_i + 1$, $SC_{q+2} = \sum_{k=1}^{q+1} T_k = \sum_{a_i \in B} a_i + 1 = \frac{1}{2} \sum_{i=1}^{p} a_i + 1$.

$\{\Leftarrow\}$ Let us say $P_{k_1} = p+1$, $P_{k_2} = p+1$, $k_1 < k_2$. So $SC_{k_1} = \frac{1}{2} \sum_{i=1}^{p} a_i + 1$, $SC_{k_2} = \frac{1}{2} \sum_{i=1}^{p} a_i + 1$. Now, using (7.4), $C = SC_{k_1} + SC_{k_2} = \sum_{i=1}^{p} a_i + 2 = \sum_{k=1}^{p+2} s_{P_k}$, while, using (7.3), $C = \sum_{k=1}^{p+2} T_k$. Hence $T_k \geq s_{P_k}$, $k = 1, \ldots, p+2$, implies $T_k = s_{P_k}$, $k = 1, \ldots, p+2$. Take $B = \{a_{P_k} : k_1 < k < k_2\}$. Then $\sum_{a_i \in B} a_i = \sum_{k=k_1+1}^{k_2-1} T_k = SC_{k_2} - T_{k_1} = \frac{1}{2} \sum_{i=1}^{p} a_i$, $\sum_{a_i \in A \setminus B} a_i = \sum_{k=k_2+1}^{p+2} T_k + \sum_{k=1}^{k_1-1} T_k = SC_{k_1} - T_{k_2} = \frac{1}{2} \sum_{i=1}^{p} a_i$.

We now show that there exists a feasible polling scheme (P, T), such that $m_i = 1$, $i = 1, \ldots, p$, $m_{p+1} = 2$, $SC_k = \frac{1}{2} \sum_{i=1}^{p} a_i + 1$ for both k with $P_k = p+1$, iff λ_i, β_i, s_i, c_i, δ_i, ϵ_i, $i = 1, \ldots, n$, and r as defined in (7.35), constitute a 'yes' instance for problem TABLE.

$\{\Rightarrow\}$ As before $C = \sum_{i=1}^{p} a_i + 2$. So for (P, T) the value of the objective function is

$$\frac{1}{2C} \sum_{i=1}^{n} c_i \lambda_i (1 + \rho_i) \sum_{\{k : P_k = i\}} SC_k^2 =$$

$$\frac{1}{2\left(\sum_{i=1}^{p} a_i + 2\right)} \left\{ \sum_{i=1}^{p} a_i \left(\sum_{i=1}^{p} a_i + 2 \right)^2 + 8 \left(\frac{1}{2} \sum_{i=1}^{p} a_i + 1 \right)^2 \right\} =$$

$$\frac{1}{2} \left(\sum_{i=1}^{p} a_i + 2 \right)^2 = r.$$

$\{\Leftarrow\}$ Problem (I) with λ_i, β_i, s_i, c_i, δ_i, ϵ_i, $i = 1, \ldots, n$, and r as defined in (7.35) has a solution which is feasible and for which the value of the objective function is not larger than r. Using the Lagrangean multiplier technique and the argumentation preceding the formulation of problem (II), it is easily verified that the value of the objective function is not larger than $\frac{1}{2} \left(\sum_{i=1}^{p} a_i + 2 \right)^2$ only for (P, T)

with $m_i = R$, $i = 1, \ldots, p$, $m_{p+1} = 2R$, $SC_k = C/m_{P_k}$, $C = \sum_{i=1}^{p} m_i a_i + m_{p+1}$, which for $R = 1$ yields the result that we have in view.

□

7.B The Golden Ratio procedure

Calculate the numbers $g(k) = k\phi^{-1} \bmod 1$ with $\phi^{-1} = \frac{1}{2}(\sqrt{5} - 1) \approx 0.618034$, the so-called Golden Ratio, $k = 1, \ldots, m$.
Let the numbers $g(k)$ with $\sum_{j=1}^{i-1} m_j + 1 \leq k \leq \sum_{j=1}^{i} m_j$ correspond to the visits to Q_i, $i = 1, \ldots, n$.
Put the numbers $g(k)$, $k = 1, \ldots, m$, in increasing order.
Let the l-th smallest number correspond to the l-th position in P, $l = 1, \ldots, m$.
Formally, $P_{\pi(k)} := i$ for k with $\sum_{j=1}^{i-1} m_j + 1 \leq k \leq \sum_{j=1}^{i} m_j$, π representing the permutation such that $g(k) \leq g(l) \Longleftrightarrow \pi(k) \leq \pi(l)$, $k, l = 1, \ldots, m$.

7.C A procedure based on extremal splittings

Before we give a detailed description, we first sketch the main motivation behind the procedure. Recall that we want to construct a polling table P that approximately minimizes (7.9) for fixed m_i and fixed $T_k = D^*_{P_k}$.
On the one hand, as seen from the argumentation preceding the formulation of problem (II), if the visits to the various queues are perfectly evenly spaced, then the polling table is optimal. In fact substituting $SC_k = \sum_{i=1}^{n} m_i D_i^* / m_{P_k}$ into (7.9) yields a lower bound for the value of (7.9) for the optimal table. On the other hand, if the visits to the various queues are perfectly evenly spaced, then the polling table obviously satisfies the following property: between any two consecutive visits to Q_i there is exactly one visit to every Q_j with $m_i = m_j$, $i \neq j$. For brevity let us refer to this property as property (E). The reverse statement does not hold. Even if the polling table satisfies property (E), then for arbitrary parameter choices the subcycle times may still be arbitrarily far from equal, and the value of (7.9) may still be arbitrarily far from minimal. Nevertheless, if the polling table satisfies property (E), then for reasonable parameter choices the visits are likely to be reasonably evenly spaced, and the polling table is likely to be reasonably good. We therefore use property (E) as the main guideline in constructing a polling table.

Let $M = \{m_i : i \in \{1, \ldots, n\}\}$ be the set of visit numbers that occur. Let $I^{(r)} = \{i \in \{1, \ldots, n\} : m_i = r\}$ be the set of the queues with common visit number r for $r \in M$.

Suppose that one has already constructed a subtable P of size $|P|$ for all the visits to the queues $i \in I^{(r_1)}, \ldots, I^{(r_q)}$ with common visit numbers $r_1, \ldots, r_q \in M$. Initially, $|P| = 0$, $\{r_1, \ldots, r_q\} = \emptyset$.
If $M \backslash \{r_1, \ldots, r_q\} \neq \emptyset$, then select a visit number r from $M \backslash \{r_1, \ldots, r_q\}$. Construct a subtable $Q^{(r)}$ of size $|Q^{(r)}| = r \times |I^{(r)}|$ for all the visits to the queues $i \in I^{(r)}$ by just concatenating r times an arbitrary sequence of the queues $i \in I^{(r)}$. Formally, with $i_1, \ldots, i_{|I^{(r)}|}$ an arbitrary sequence of the queues $i \in I^{(r)}$, $Q^{(r)}_{j+(k-1) \times |I^{(r)}|} := i_j$ for $j = 1, \ldots, |I^{(r)}|$, $k = 1, \ldots, r$. Obviously $Q^{(r)}$ satisfies property (E).
Construct subsequently a subtable $P^{(r)}$ of size $|P^{(r)}| = |P| + |Q^{(r)}|$, by inserting the visits from the subtable $Q^{(r)}$ in the subtable P in the following manner. Put the visits from $Q^{(r)}$ at positions in $P^{(r)}$ as evenly spaced as possible, i.e, put the visit at the k-th position in $Q^{(r)}$ at the $(k + d(k))$-th position in $P^{(r)}$, $k = 1, \ldots, |Q^{(r)}|$. Here $d(k) = \left[(k-1) \times \frac{|P|}{|Q^{(r)}|}\right]$, $k = 1, \ldots, |Q^{(r)}|$, with $[x]$ denoting the nearest integer to x. Put the visits from P at the remaining positions in $P^{(r)}$, i.e., put the visit at the $\chi(l + l_0)$-th position in P at the l-th position in $P^{(r)}$ that is not yet occupied by a visit from $Q^{(r)}$, $l = 1, \ldots, |P|$. Here $\chi(k) = ((k-1) \bmod |P|) + 1$. Choose l_0 from $\{1, \ldots, |P|\}$ such that the objective function (7.9) properly applied to $P^{(r)}$ is minimal. Formally, $P^{(r)}_{k+d(k)} := Q^{(r)}_k$, $k = 1, \ldots, |Q^{(r)}|$, $P^{(r)}_{k+l} := P_{\chi(l+l_0)}$, $k = 1, \ldots, |Q^{(r)}|$, $l = d(k)+1, \ldots, d(k+1)$. Thus in $P^{(r)}$ the number of visits from P between the k-th and $(k+1)$-th visit from $Q^{(r)}$ equals $d(k+1) - d(k) = \left[k \times \frac{|P|}{|Q^{(r)}|}\right] - \left[(k-1) \times \frac{|P|}{|Q^{(r)}|}\right]$, $k = 1, \ldots, |Q^{(r)}|$. This distancing is closely related to extremal splittings of point processes, cf. Hajek [116]. Note that the internal visit order from P and $Q^{(r)}$ is maintained. Hence, by induction, $P^{(r)}$ satisfies property (E).
Repeat with P replaced by $P^{(r)}$. Finally $|P| = m$, $\{r_1, \ldots, r_q\} = M$.

Remark 7.C.1 The above-described method appears to be similar to an algorithm presented in Arian & Levy [12]. Simulation results in [12] suggest that the visit numbers r can best be selected $M \backslash \{r_1, \ldots, r_q\}$ in descending order. □

7.D Proof of Lemma 7.4.1

Lemma 7.4.1
The set of linear equations

$$T_k = \rho_{P_k}(1 + \epsilon_{P_k}) \sum_{l=1}^{m} h_{kl} T_l + (1 + \delta_{P_k}) s_{P_k}, \qquad k = 1, \ldots, m, \tag{7.36}$$

has a unique solution; this solution is non-negative.

Proof

Define the matrix A by $A_{kl} := \rho_{P_k}(1+\epsilon_{P_k})h_{kl}$, $k,l = 1,\ldots,m$, and the vector b by $b_k := (1+\delta_{P_k})s_{P_k}$, $k = 1,\ldots,m$. Then the set of linear equations (7.36) may be rewritten as $(I-A)T = b$.

As A is a non-negative irreducible matrix, A has a real eigenvalue μ, which is strictly maximal in absolute value, cf. Seneta [163] pp. 3-4. For μ holds $\min_{1\le l\le m} \sum_{k=1}^{m} A_{kl} \le \mu \le \max_{1\le l\le m} \sum_{k=1}^{m} A_{kl}$, cf. [163] p. 8. $\sum_{k=1}^{m} A_{kl} = \sum_{i=1}^{n} \sum_{\{k:P_k=i\}} \rho_{P_k}(1+\epsilon_{P_k})\, h_{kl} = \rho + \sum_{i=1}^{n} \epsilon_i\, \rho_i$, as $\sum_{\{k:P_k=i\}} h_{kl} = 1$, $l = 1,\ldots,m$, so $\mu = \rho + \sum_{i=1}^{n} \epsilon_i\, \rho_i$. As b is a non-negative vector, $\rho + \sum_{i=1}^{n} \epsilon_i\rho_i < 1$ implies that $(I-A)T = b$ has a unique solution $T = (I-A)^{-1}b \ge 0$, cf. [163] p. 30.

□

Chapter 8

Optimal allocation of customer types to servers

8.1 Introduction

In the present chapter we consider a system in which there are several parallel servers to process jobs generated at several distinct sources. Such a system arises quite naturally in modeling situations where a pool of resources is available to perform various kinds of activities. Examples may be found in distributed computer systems, flexible manufacturing systems, and telecommunication networks. Another example may be found in a situation where a pool of repair crews is available to perform maintenance activities at various installations.

In such a system, in which there are several servers to process jobs generated at several sources, usually some freedom of decision exists as to which server is to process which job at what time. Thus the need arises for a scheduling strategy, i.e., a collection of decision instructions for scheduling the jobs. At a global level decisions need to be made about which server is to process which job. Subsequently at a local level decisions need to be made about the order of service. In this chapter we mainly focus on the global scheduling problem; we hardly touch on the local scheduling problem. Locally, the order of service is assumed not to discriminate between the sources from which the jobs originated.

The main function of a global scheduling strategy is load sharing; a strategy should make the servers cooperate in sharing the load of the system so as to optimize the system performance. Load sharing is also frequently referred to as load balancing. The term load balancing arises from the intuition that to optimize the system performance the load should be balanced among the servers. In this chapter we find however that the load, in the sense of traffic intensity,

in general should *not* be completely balanced.

Wang & Morris [186] give a comprehensive survey of the overwhelming variety of approaches to load sharing in the literature. They identify some fundamentally distinguishing features of load sharing strategies. A first distinction refers to the side that takes the initiative in scheduling the jobs. In source-initiative policies a source decides to which server to route a job. In server-initiative policies a server decides from which source to get a job. Consequently in source-initiative policies decisions are typically made at arrival epochs, whereas in server-initiative policies decisions are typically made at service completion epochs (possibly at arrival epochs when servers are idle). Moreover, in source-initiative policies queues tend to form at the servers, whereas in server-initiative policies queues tend to form at the sources. Of course there are also policies conceivable in which both the sources and the servers participate in allocating the jobs.

A second distinction refers to the amount of information that is used in allocating the jobs. In purely static policies only information is used about the basic characteristics of the system, like the traffic intensities. In dynamic policies also information is used about the actual state of the system, like the queue lengths. Evidently, in principle the performance of the system may improve substantially by using such information in allocating the jobs. However, gathering such information and implementing a sophisticated dynamic allocation strategy may involve a considerable communication overhead and complicate the operation of the system significantly. Therefore dynamic policies are not necessarily preferable to static policies.

In this chapter we assume that customers are allocated to the servers in a probabilistic manner; upon arrival customers are sent to one of the servers according to a matrix of routing probabilities. Such a load sharing strategy is commonly referred to as random splitting. In the taxonomy of Wang & Morris [186] random splitting belongs to the class of static source-initiative load sharing strategies. We are interested in the problem of finding a random splitting that minimizes a weighted sum of the mean waiting times.

The novelty of the model lies in the combination of heterogeneous servers (i.e. different service rates), heterogeneous sources (i.e. different service times), and a fairly general cost function. Buzacott & Shanthikumar [64] consider a version of the problem with homogeneous servers and the overall mean waiting time as performance measure, which we will discuss later on in greater technical detail. Buzen & Chen [65] consider a variant of the problem with a single source and the overall mean sojourn time as performance criterion. A natural approach to deal with heterogeneous servers and heterogeneous sources might be to aggregate the different sources into a single source, and then use the results of Buzen & Chen. Each server would thus handle a traffic mix of the same, heterogeneous, composition, but of possibly different intensity, depending on the processing rate of the servers. In this chapter we find however that *each server should handle a traffic mix as homogeneous as possible.* For homogeneous sources Boxma & Combé [41] show that 'pattern' allocation outperforms

probabilistic allocation, but that the optimal routing probabilities of Buzen & Chen provide a reasonable indication for the optimal occurrence fractions in 'pattern' allocation. It is likely that similar observations hold for heterogeneous sources.

Tantawi & Towsley [178], [179] and De Souza e Silva & Gerla [81] consider optimal load balancing models of distributed computer systems consisting of a number of heterogeneous host computers connected by a communication network. A job may be either processed at the host to which it arrives or transferred to another host. In the latter case, a transferred job incurs a communication delay in addition to the queueing delay at the host on which it is processed. The assumptions in [178], [179], and [81] on the service requirements of jobs are however somewhat restrictive.

In some situations it may be desirable that job classes are not split among different servers. In flexible manufacturing systems e.g. such a splitting may be undesirable because handling a particular job class usually requires special expensive tools. In case job classes are not split among different servers, a load sharing strategy is commonly referred to as source partitioning. In view of the practical relevance we are specifically interested in finding an optimal source partitioning. For general partitioning problems Anily & Federgruen [10] identify analytical properties of the cost function under which an optimal solution has a simple structure, thus allowing for a simple solution method. They mention the problem of finding an optimal source partitioning as an example for which the mean number of waiting customers as cost function (which by Little's law is nothing but a specific weighted sum of the mean waiting times) does *not* have such analytical properties.

The remainder of the chapter is organized as follows. In Section 8.2 we present a detailed model description. We then consider the problem of finding an optimal random splitting. In Section 8.3 we expose the structure of an optimal allocation and in Section 8.4 we describe for some special cases in detail how the structure may be exploited in actually determining an optimal allocation. In Section 8.5 we consider the problem of finding an optimal source partitioning. We show the problem to be NP-hard and indicate how the structure of an optimal non-deterministic allocation may be used as a heuristic guideline in searching for an optimal deterministic allocation. In Section 8.6 we conclude with some remarks and suggestions for further research.

8.2 Model description

The model under consideration consists of n customer types attended by m parallel non-identical servers. Customers arrive according to Poisson processes. The arrival rate of type-i customers is λ_i, $i = 1, \ldots, n$. The total arrival rate is $\lambda := \sum_{i=1}^{n} \lambda_i$. Upon arrival customers are routed to one of the servers. Type-i customers are routed to server j with probability x_{ij}, $i = 1, \ldots, n$, $j = 1, \ldots, m$. The matrix $x = (x_{ij})$ of routing probabilities will be referred to as the (prob-

abilistic) allocation of the customer types to the servers. In case the matrix of routing probabilities is 0-1 the (probabilistic) allocation will be referred to as deterministic. When processed at server j, type-i customers require service times having distribution $F_{ij}(t) = F_i(\mu_j t)$, i.e., type-i customers require amounts of service having distribution $F_i(t)$, while server j has processing rate μ_j. In other words, the servers may have different characteristics, the customer types may have different characteristics, but servers cannot 'specialize' in some of the customer types. Denote by β_i and $\beta_i^{(2)}$ the first and second moment of $F_i(t)$, $i = 1, \ldots, n$. We assume $F_i(0) < 1$, so $\beta_i > 0$, $\beta_i^{(2)} > 0$, $i = 1, \ldots, n$. Define the traffic intensity associated with type-i customers as $\rho_i := \lambda_i \beta_i$, $i = 1, \ldots, n$. The total traffic intensity is $\rho := \sum_{i=1}^{n} \rho_i$. The order of service is assumed not to discriminate between the various customer types. Further all arrival, service, and routing processes are assumed to be mutually independent.

The queues that form at the servers are ordinary $M/G/1$ queues. The arrival rate at server j is $\sum_{i=1}^{n} x_{ij}\lambda_i$. Customers that are routed to server j require service times having distribution $\sum_{i=1}^{n} x_{ij}\lambda_i F_i(\mu_j t) / \sum_{i=1}^{n} x_{ij}\lambda_i$ with first moment $\left[\sum_{i=1}^{n} x_{ij}\lambda_i\beta_i\right] \Big/ \left[\mu_j \sum_{i=1}^{n} x_{ij}\lambda_i\right]$ and second moment $\left[\sum_{i=1}^{n} x_{ij}\lambda_i\beta_i^{(2)}\right] \Big/ \left[\mu_j^2 \sum_{i=1}^{n} x_{ij}\lambda_i\right]$, $j = 1, \ldots, m$. Define the traffic intensity at server j as $\sum_{i=1}^{n} x_{ij}\lambda_i\beta_i$, $j = 1, \ldots, m$. Necessary and sufficient ergodicity conditions are

$$\sum_{i=1}^{n} x_{ij}\lambda_i\beta_i < \mu_j, \qquad j = 1, \ldots, m. \tag{8.1}$$

Denote by $\mu := \sum_{j=1}^{m} \mu_j$ the total processing rate of the servers. Summing (8.1) with respect to $j = 1, \ldots, m$ yields $\rho < \mu$, as $\sum_{j=1}^{m} x_{ij} = 1$, $i = 1, \ldots, n$. Throughout the chapter $\rho < \mu$ is assumed to hold.

We are interested in the problem of finding an allocation that minimizes a weighted sum of the mean waiting times. Therefore we first derive a formula that describes the mean waiting times as function of the allocation matrix $x = (x_{ij})$. Denote by $\mathbf{W}_i$ the waiting time of an arbitrary type-i customer, $i = 1, \ldots, n$, i.e., the time from its arrival to the start of its service. Denote by $\mathbf{V}_j$ the waiting time of an arbitrary customer that is routed to server j, $j = 1, \ldots, m$.

As the order of service is assumed not to discriminate between the various customer types,

$$\mathrm{E}\mathbf{W}_i = \sum_{j=1}^{m} x_{ij}\mathrm{E}\mathbf{V}_j, \qquad i = 1, \ldots, n. \tag{8.2}$$

As the queues that form at the servers are ordinary $M/G/1$ queues,

$$\mathrm{E}\mathbf{V}_j = \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x_{ij}}{2\mu_j \left(\mu_j - \sum_{i=1}^{n} \lambda_i \beta_i x_{ij}\right)}, \qquad j = 1, \ldots, m. \tag{8.3}$$

Let c_i represent the waiting cost per unit of time of a type-i customer, $i = 1, \ldots, n$. We assume $c_i > 0$, $i = 1, \ldots, n$. The mean total waiting cost per unit of time amounts to $\sum_{i=1}^{n} c_i \lambda_i \mathrm{E}\mathbf{W}_i$. Using (8.2) and (8.3),

$$\sum_{i=1}^{n} c_i \lambda_i \mathrm{E}\mathbf{W}_i = \sum_{j=1}^{m} \frac{\left(\sum_{i=1}^{n} \lambda_i c_i x_{ij}\right)\left(\sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x_{ij}\right)}{2\mu_j \left(\mu_j - \sum_{i=1}^{n} \lambda_i \beta_i x_{ij}\right)}. \tag{8.4}$$

8.3 Finding an optimal random splitting

In this section we consider the problem of finding an optimal random splitting, i.e., a probabilistic allocation of the customer types to the servers that minimizes the mean total waiting cost per unit of time. Using (8.1) and (8.4), we formulate the problem as follows.

Problem (I).

$$\text{minimize} \qquad f(x) = \sum_{j=1}^{m} \frac{\left(\sum_{i=1}^{n} \lambda_i c_i x_{ij}\right)\left(\sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x_{ij}\right)}{\mu_j \left(\mu_j - \sum_{i=1}^{n} \lambda_i \beta_i x_{ij}\right)} \tag{8.5}$$

$$\text{subject to} \qquad \sum_{i=1}^{n} \lambda_i \beta_i x_{ij} < \mu_j, \qquad j = 1, \ldots, m;$$

$$\sum_{j=1}^{m} x_{ij} = 1, \qquad i = 1, \ldots, n;$$

$$x_{ij} \geq 0, \qquad i = 1, \ldots, n,\ j = 1, \ldots, m.$$

Problem (I) is a non-linear programming problem. It is easily verified that the objective function $f(\cdot)$ is not convex, so that it is not guaranteed that there

exists a unique Kuhn-Tucker point. Moreover, finding a Kuhn-Tucker point is not quite straightforward.
All in all there is not an obvious way of solving problem (I). Nevertheless, if one is purely interested in computing an optimal allocation for some given parameters, then one might in principle proceed to solving problem (I) by standard non-convex programming techniques. That is however not what we are interested in here. What we are primarily interested in, is obtaining some insight into the structural properties of an optimal allocation. We will show that an optimal solution of problem (I) indeed exhibits a very characteristic structure. As secondary motivation, the structural properties do not only provide some insight, but are also very useful in computing an optimal allocation. Specifically we will describe in Section 8.4 how in cases with identical servers where all the customer types are in a sense ordered, the structure may be exploited in a very simple manner in actually determining an optimal solution of problem (I). In these cases there exists a unique Kuhn-Tucker point exhibiting the structure of an optimal solution. So it is guaranteed that this Kuhn-Tucker point *is* the optimal solution. Moreover, finding this Kuhn-Tucker point is comparatively straightforward in these cases. In cases where not all the customer types are ordered, the structure may still be exploited in actually determining an optimal solution of problem (I), but not in a manner as simple. In Section 8.5 we indicate how the knowledge of the structure of an optimal solution of problem (I) may also be used as a guideline in heuristically solving the NP-hard *integer* version (integer x_{ij}'s) of problem (I).

We now expose the structure of an optimal allocation x^*. We first introduce some notation. For a given allocation x, define $K_j(x) := \{i \mid x_{ij} > 0\}$ to be the index set of the customer types (partially) allocated to server j. Define $A_j(x) := \left\{ \left(\frac{c_i}{\beta_i}, \frac{\beta_i^{(2)}}{\beta_i} \right) \mid i \in K_j(x) \right\}$ to be the set of $\left(\frac{c_i}{\beta_i}, \frac{\beta_i^{(2)}}{\beta_i} \right)$-values corresponding to the customer types allocated to server j. Denote $P_j(x) := \text{int}(\text{conv}(A_j(x)))$, with $\text{int}(\text{conv}(\cdot))$ denoting the interior of the convex hull. The set $P_j(x)$ may be interpreted as the global range of $\left(\frac{c_i}{\beta_i}, \frac{\beta_i^{(2)}}{\beta_i} \right)$-values corresponding to the customer types allocated to server j. Denote

$$B_j(x) := \left[\sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x_{ij} \right] \Big/ \left[\mu_j \left(\mu_j - \sum_{i=1}^{n} \lambda_i \beta_i x_{ij} \right) \right],$$

$$C_j(x) := \left[\sum_{i=1}^{n} \lambda_i c_i x_{ij} \right] \Big/ \left[\mu_j \left(\mu_j - \sum_{i=1}^{n} \lambda_i \beta_i x_{ij} \right) \right].$$

The numbers $B_j(x)$ and $C_j(x)$ may be interpreted as measures for the '$\beta_i^{(2)}/\beta_i$-weight' and the 'c_i/β_i-weight' associated with the customer types allocated to server j.

We now expose the structure of an optimal allocation x^* in terms of the corresponding sets $P_j(x^*)$. Intuitively, it is to be expected that an optimal allocation will satisfy one of the following two (in general mutually exclusive) 'extremal' properties.
(i). Each server handles a traffic mix of the same composition (e.g. $x^*_{ij} = \mu_j/\mu$, $i = 1, \ldots, n$, $j = 1, \ldots, m$), so that the traffic mix at each server is completely heterogeneous;
(ii). Each server handles a traffic mix as homogeneous as possible, so that different servers deal with traffic mixes of a completely different composition.
The next Lemma says that an optimal allocation in fact satisfies the second property (so the first one not in general).

Lemma 8.3.1
$P_{j'}(x^*) \cap P_{j''}(x^*) = \emptyset$ for $j' \neq j''$.
In other words, if $\left(\frac{c_i}{\beta_i}, \frac{\beta_i^{(2)}}{\beta_i}\right) \in P_j(x^*)$, then $x^*_{ij} = 1$.

Proof
See Appendix 8.A.

□

Lemma 8.3.1 suggests that the customer types should be clustered according to the corresponding $\left(\frac{c_i}{\beta_i}, \frac{\beta_i^{(2)}}{\beta_i}\right)$-values. As a consequence, different servers will deal with different traffic mixes. The next Lemma says that different traffic mixes may however not involve an arbitrarily different '$\beta_i^{(2)}/\beta_i$-weight' and 'c_i/β_i-weight'.

Lemma 8.3.2
If $B_{j'}(x^*) \geq B_{j''}(x^*)$, $C_{j'}(x^*) \geq C_{j''}(x^*)$,
then $\mu_{j'} B_{j'}(x^*) C_{j'}(x^*) \leq \mu_{j''} B_{j''}(x^*) C_{j''}(x^*)$.

Proof
See Appendix 8.B.

□

Lemma 8.3.2 states that if one server carries both larger $B_j(x^*)$ and $C_j(x^*)$ than another, then it cannot carry larger $\mu_j B_j(x^*) C_j(x^*)$ as well.

In the remainder of this section as well as in the next section we consider the case of identical servers, i.e., $\mu_j = \mu/m$, $j = 1, \ldots, m$. Lemma 8.3.2 then states that it is no longer possible that one server carries both larger $B_j(x^*)$ and $C_j(x^*)$ than another.

Corollary 8.3.1
$B_{j'}(x^*) \geq B_{j''}(x^*) \Longleftrightarrow C_{j'}(x^*) \leq C_{j''}(x^*)$.

We now assume that the servers are indexed such that $B_{j'}(x^*) \geq B_{j''}(x^*)$, $C_{j'}(x^*) \leq C_{j''}(x^*)$ for $j' < j''$.

Lemma 8.3.3
Assume $x^*_{i'j'} > 0$, $x^*_{i''j''} > 0$.
If $\frac{c_{i'}}{\beta_{i'}} \leq \frac{c_{i''}}{\beta_{i''}}$, $\frac{\beta^{(2)}_{i'}}{\beta_{i'}} \geq \frac{\beta^{(2)}_{i''}}{\beta_{i''}}$, $\left(\frac{c_{i'}}{\beta_{i'}}, \frac{\beta^{(2)}_{i'}}{\beta_{i'}}\right) \neq \left(\frac{c_{i''}}{\beta_{i''}}, \frac{\beta^{(2)}_{i''}}{\beta_{i''}}\right)$, then $j' \leq j''$.

Proof
See Appendix 8.C.

□

Lemma 8.3.3 states that expensive, calm (cheap, wild) customer types with large (small) c_i/β_i and small (large) $\beta^{(2)}_i/\beta_i$ should be sent to servers with small (large) $B_j(x^*)$ and large (small) $C_j(x^*)$, thus experiencing a small (large) waiting time. (Note that $B_j(x)$ is in fact twice the mean waiting time at server j for allocation x.) Lemma 8.3.3 does however not indicate what should be done with expensive but wild (cheap but calm) customer types with large (small) c_i/β_i and large (small) $\beta^{(2)}_i/\beta_i$. Indeed, it depends not only on their own individual c_i/β_i and $\beta^{(2)}_i/\beta_i$ but also on some other less seizable factors whether they should be sent to servers with small $B_j(x^*)$ and large $C_j(x^*)$ or with large $B_j(x^*)$ and small $C_j(x^*)$.
Lemma 8.3.3 allows us to strengthen the statements on the clustering of the customer types in Lemma 8.3.1. We first introduce some additional notation. Define

$$Q_j(x) := \bigcup_{i \in K_j(x)} \left\{ (y,z) : y \leq \frac{c_i}{\beta_i}, z \geq \frac{\beta^{(2)}_i}{\beta_i}, (y,z) \neq \left(\frac{c_i}{\beta_i}, \frac{\beta^{(2)}_i}{\beta_i}\right) \right\},$$

$$R_j(x) := \bigcup_{i \in K_j(x)} \left\{ (y,z) : y \geq \frac{c_i}{\beta_i}, z \leq \frac{\beta^{(2)}_i}{\beta_i}, (y,z) \neq \left(\frac{c_i}{\beta_i}, \frac{\beta^{(2)}_i}{\beta_i}\right) \right\}.$$

Denote $S_j(x) := Q_j(x) \cap R_j(x)$, $T_j(x) := P_j(x) \cup S_j(x)$.

Lemma 8.3.4
$S_{j'}(x^*) \cap S_{j''}(x^*) = \emptyset$ for $j' \neq j''$.
In other words, if $\left(\frac{c_i}{\beta_i}, \frac{\beta^{(2)}_i}{\beta_i}\right) \in S_j(x^*)$, then $x^*_{ij} = 1$.

Proof
See Appendix 8.D.

□

Lemma 8.3.4 suggests that in the case of identical servers the customer types should be clustered according to the corresponding $\left(\frac{c_i}{\beta_i}, \frac{\beta_i^{(2)}}{\beta_i}\right)$-values in an even stronger sense than stated before in Lemma 8.3.1.

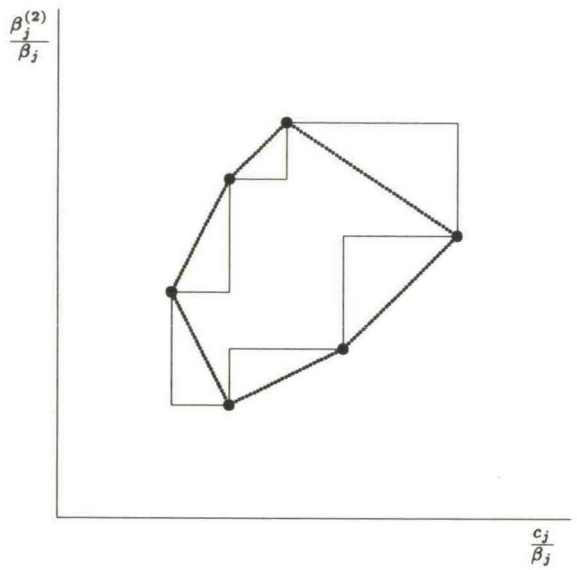

FIGURE 8.1. The sets $P_i(x)$ and $S_i(x)$.

It is easily verified from the definition of $P_j(x)$ and $S_j(x)$ that if $P_{j'}(x) \cap P_{j''}(x) = \emptyset$ and $S_{j'}(x) \cap S_{j''}(x) = \emptyset$, then also $P_{j'}(x) \cap S_{j''}(x) = \emptyset$, i.e., $T_{j'}(x) \cap T_{j''}(x) = \emptyset$, see Figure 8.1, where the bold dots constitute the set $A_j(x)$. The area inside the dotted lines corresponds to the set $P_j(x)$. The rectangular area represents the set $S_j(x)$. So in the case of identical servers Lemma 8.3.1 and Lemma 8.3.4 may be summarized as follows.

Corollary 8.3.2
$T_{j'}(x^*) \cap T_{j''}(x^*) = \emptyset$ for $j' \neq j''$.
In other words, if $\left(\frac{c_i}{\beta_i}, \frac{\beta_i^{(2)}}{\beta_i}\right) \in T_j(x^*)$, then $x_{ij}^* = 1$.

The optimality of clustering as exposed in the previous Lemma's suggests that the optimal routing probabilities are almost all equal to either 0 or 1. Although the settings are quite different, the latter observation strongly reminds of the

vertex-allocation theorem for the optimal routing of single customer chains in closed product-form networks, saying that each customer should consistently select the same server for each request type rather than choose probabilistically, cf. Tripathi & Woodside [180], Woodside & Tripathi [189], Cheng & Muntz [67].

In the next section we show how the structure, as characterized in the previous Lemma's, may be exploited in computing an optimal allocation.

8.4 The case of ordered customer types

In this section we show how the structure, as characterized in the Lemma's of the previous section, may be exploited in computing an optimal allocation. We make the following assumption.

Assumption 8.4.1
The customer types are ordered such that

$$\frac{c_{i'}}{\beta_{i'}} \leq \frac{c_{i''}}{\beta_{i''}}, \frac{\beta_{i'}^{(2)}}{\beta_{i'}} \geq \frac{\beta_{i''}^{(2)}}{\beta_{i''}}, \left(\frac{c_{i'}}{\beta_{i'}}, \frac{\beta_{i'}^{(2)}}{\beta_{i'}}\right) \neq \left(\frac{c_{i''}}{\beta_{i''}}, \frac{\beta_{i''}^{(2)}}{\beta_{i''}}\right) \text{ for } i' < i''.$$

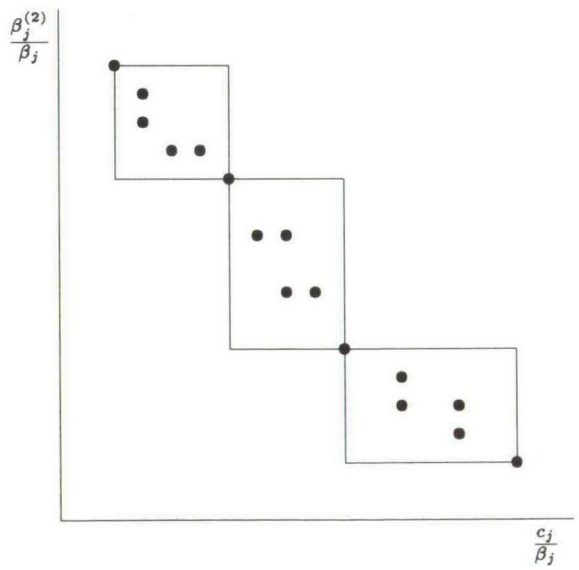

FIGURE 8.2. The case of ordered customer types.

Below we describe in detail how the structure may be exploited in actually determining an optimal allocation in cases where the customer types are ordered in the sense of Assumption 8.4.1. If Assumption 8.4.1 is satisfied, then Corollary 8.3.2 provides a very strong characterization of an optimal allocation, see Figure 8.2, where the rectangles represent the sets $T_j(x)$ for an instance with $m = 3$ servers and $n = 16$ customer types. The fact that the sets $T_i(x)$ do

not intersect, completely determines their 'position', so that only the problem remains to determine their 'size'. In cases where the customer types are not ordered, the structure may still be exploited in actually determining an optimal allocation but not in a manner as simple.

Theoretically speaking, Assumption 8.4.1 is somewhat restrictive. However, there are several cases of practical interest that satisfy Assumption 8.4.1.

Case i. $c_i/\beta_i = \gamma$, $i = 1, \ldots, n$.
In other words, the waiting costs per unit of time are proportional to the mean service times. This is the case when the goal is minimizing the mean amount of waiting work, $\sum_{i=1}^{n} \rho_i \mathrm{EW}_i$. Minimizing the mean amount of waiting work is equivalent to minimizing the mean total amount of work, as the difference, the mean amount of work in service always equals $\frac{1}{2}\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}$, irrespective of the allocation x. When the customer types have the same mean service time, minimizing the mean amount of waiting work is also equivalent to minimizing the overall mean waiting time.
For $c_i/\beta_i = \gamma$, $i = 1, \ldots, n$, Corollary 8.3.1 reduces to

$$\frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x_{ij'}^*}{\mu/m - \sum_{i=1}^{n} \lambda_i \beta_i x_{ij'}^*} \geq \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x_{ij''}^*}{\mu/m - \sum_{i=1}^{n} \lambda_i \beta_i x_{ij''}^*} \iff \sum_{i=1}^{n} \lambda_i \beta_i x_{ij'}^* \leq \sum_{i=1}^{n} \lambda_i \beta_i x_{ij''}^*.$$

In particular,

$$\sum_{i=1}^{n} \lambda_i \beta_i x_{ij'}^* = \sum_{i=1}^{n} \lambda_i \beta_i x_{ij''}^* \iff \sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x_{ij'}^* = \sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x_{ij''}^*. \tag{8.6}$$

For $c_i/\beta_i = \gamma$, $i = 1, \ldots, n$, the set $T_j(x)$ reduces to the line-segment

$$\left\{ (\gamma, z) \mid \min_{i \in K_j(x)} \beta_i^{(2)}/\beta_i < z < \max_{i \in K_j(x)} \beta_i^{(2)}/\beta_i \right\}.$$

Thus Corollary 8.3.2 says that the customer types should be clustered according to the corresponding $\beta_i^{(2)}/\beta_i$-values. This means that

$$\left(\sum_{i=1}^{n} \lambda_i \beta_i x_{ij}^*, \sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x_{ij}^* \right) = \left(\frac{1}{m} \sum_{i=1}^{n} \lambda_i \beta_i, \frac{1}{m} \sum_{i=1}^{n} \lambda_i \beta_i^{(2)} \right),$$

$j = 1, \ldots, m$, will only hold when all the customer types do not only have the same $c_i/\beta_i = \gamma$, but also happen to have the same $\beta_i^{(2)}/\beta_i$, which in general is not the case. In view of (8.6) we may thus conclude that in general $\sum_{i=1}^{n} \lambda_i \beta_i x_{ij'}^* \neq \sum_{i=1}^{n} \lambda_i \beta_i x_{ij''}^*$, i.e., the total load should *not* be completely balanced.

Case ii. $\beta_i^{(2)}/\beta_i = \delta$, $i = 1, \ldots, n$.
In other words, the mean residual service times are constant. This is the case when the customer types have the same service time characteristics, but different priorities reflected in different waiting costs per unit of time.
For $\beta_i^{(2)}/\beta_i = \delta$, $i = 1, \ldots, n$, Corollary 8.3.1 reduces to

$$\sum_{i=1}^{n} \lambda_i \beta_i x_{ij'}^* \geq \sum_{i=1}^{n} \lambda_i \beta_i x_{ij''}^* \iff \frac{\sum_{i=1}^{n} \lambda_i c_i x_{ij'}^*}{\mu/m - \sum_{i=1}^{n} \lambda_i \beta_i x_{ij'}^*} \leq \frac{\sum_{i=1}^{n} \lambda_i c_i x_{ij''}^*}{\mu/m - \sum_{i=1}^{n} \lambda_i \beta_i x_{ij''}^*}.$$

In particular,

$$\sum_{i=1}^{n} \lambda_i \beta_i x_{ij'}^* = \sum_{i=1}^{n} \lambda_i \beta_i x_{ij''}^* \iff \sum_{i=1}^{n} \lambda_i c_i x_{ij'}^* = \sum_{i=1}^{n} \lambda_i c_i x_{ij''}^*. \tag{8.7}$$

For $\beta_i^{(2)}/\beta_i = \delta$, $i = 1, \ldots, n$, the set $T_j(x)$ reduces to the line-segment

$$\left\{ (y, \delta) \mid \min_{i \in K_j(x)} c_i/\beta_i < y < \max_{i \in K_j(x)} c_i/\beta_i \right\}.$$

Thus Corollary 8.3.2 says that the customer types should be clustered according to the corresponding c_i/β_i-values (which strongly reminds of the $c\mu$-rule, cf. [151], although the $c\mu$-rule in fact refers to the order of service of customer types rather than the clustering of customer types). This implies that

$$\left(\sum_{i=1}^{n} \lambda_i \beta_i x_{ij}^*, \sum_{i=1}^{n} \lambda_i c_i x_{ij}^* \right) = \left(\frac{1}{m} \sum_{i=1}^{n} \lambda_i \beta_i, \frac{1}{m} \sum_{i=1}^{n} \lambda_i c_i \right),$$

$j = 1, \ldots, m$, will only hold when all the customer types do not only have the same $\beta_i^{(2)}/\beta_i = \delta$, but also happen to have the same c_i/β_i, which in general is not the case. In view of (8.7) we may thus again conclude that in general $\sum_{i=1}^{n} \lambda_i \beta_i x_{ij'}^* \neq \sum_{i=1}^{n} \lambda_i \beta_i x_{ij''}^*$, i.e., the total load should *not* be completely balanced.

Case iii. $c_i = c$, $i = 1, \ldots, n$, $\beta_{i'} \leq \beta_{i''} \iff \beta_{i'}^{(2)}/\beta_{i'} \leq \beta_{i''}^{(2)}/\beta_{i''}$.
In other words, the waiting costs per unit of time are constant. This is the case when the goal is minimizing the overall mean waiting time. Moreover, a larger mean service time corresponds to a larger mean residual service time. For several natural service time distributions this is indeed the case. Corollary 8.3.2 then says that the customer types should be clustered according to the corresponding β_j-values. Again it may verified that the total load should *not* be completely balanced, unless the values of λ_i, β_i, $\beta_i^{(2)}$ of the customer types happen to satisfy some very specific relationships.

Remark 8.4.1
Buzacott & Shanthikumar [64] consider the problem of finding an optimal allocation in the case $c_i = c = 1$, $i = 1, \ldots, n$, i.e., the goal is minimizing the mean overall waiting time. In addition they require that the total load be balanced, i.e., $r_j^* = \sum_{i=1}^{n} \rho_i x_{ij} = \rho/m$, $j = 1, \ldots, m$. They show that if the agreeability condition $\beta_{i'} \leq \beta_{i''} \iff \beta_{i'}^{(2)}/\beta_{i'} \leq \beta_{i''}^{(2)}/\beta_{i''}$ is satisfied, then the customer types should be clustered according to the corresponding β_i-values, as we also concluded in Case iii. above, *without* requiring that the total load be balanced. In fact we concluded in Case iii. that the total load should *not* be completely balanced.

□

We now describe a method for determining an optimal allocation in cases that satisfy Assumption 8.4.1. Here we sketch the main idea of the method. In Appendix 8.E we describe the method in detail.
From Lemma 8.3.3 we know that the structure of an optimal allocation is (i) $(x_{1j}^*, \ldots, x_{nj}^*) = (0, \ldots, 0, x_{i'_j j}^*, 1, \ldots, 1, x_{i''_j j}^*, 0, \ldots, 0)$, with (ii) $\sum_{k=1}^{j} x_{i'_j k}^* = 1$, $\sum_{k=j}^{m} x_{i''_j j}^* = 1$. So an optimal allocation is then completely characterized by $s_j^* = \sum_{i=1}^{n} x_{ij}^*$, $j = 1, \ldots, m$. The global idea of the method is now to perform a kind of binary search with regard to s_1^*.
Step 1. Determine a lower and an upper bound for s_1^*.
Step 2. Make an estimate s_1 for s_1^*, somewhere in between lower and upper bound.
Step 3. Given s_j, determine s_{j+1}, for $j = 1, 2, \ldots, m-1$, from the knowledge of the structure of an optimal allocation (i), (ii), together with (iii) $\dfrac{\partial f(x)}{\partial x_{i''_j j}} = \dfrac{\partial f(x)}{\partial x_{i'_{j+1} j+1}}$. The latter condition necessarily holds for an optimal allocation, as may be formally verified from the Kuhn-Tucker conditions. In Appendix 8.E we show that through (i), (ii), (iii), s_j uniquely determines s_{j+1}, $j = 1, 2, \ldots m-1$.
Step 4. Sooner or later one either runs out of servers or out of customer types. If one runs out of servers, then apparently $s_1 < s_1^*$, so then replace the old lower bound by s_1. If one runs out of customer types, then apparently $s_1 > s_1^*$, so then replace the old upper bound by s_1. Repeat the procedure until lower and upper bound are sufficiently close.

We now consider some examples illustrating that in general the load should indeed not be completely balanced. We assume that there are $m = 4$ servers and $n = 4$ customer types. We take $(\lambda_1, \lambda_2, \lambda_3, \lambda_4) = \alpha(8, 8, 1, 1)$, $(\beta_1, \beta_2, \beta_3, \beta_4) = (1, 2, 4, 8)$, so $(\rho_1, \rho_2, \rho_3, \rho_4) = 4\alpha(2, 4, 1, 2)$. We take successively $\alpha = 0.01$,

$\alpha = 0.05$, $\alpha = 0.10$, $\alpha = 0.11$, so successively $\rho = 0.36$, $\rho = 1.80$, $\rho = 3.6$, $\rho = 3.96$. We assume that $\beta_i^{(2)} = \kappa\beta_i^2$, $i = 1, \ldots, n$, so that the agreeability condition is trivially satisfied. Note that κ influences the value of the objective function $f(\cdot)$ linearly, so that an optimal allocation is independent of κ. We compare the value of the objective function $f(\cdot)$ with $\kappa = 1$ for each of the following three allocations:

(i). x^S, the completely symmetric allocation, i.e., $x_{ij}^S = 1/m$, $i = 1, \ldots, n$, $j = 1, \ldots, m$;

(ii). x^B, the optimal allocation with $r_j^B = \sum_{i=1}^{n} \rho_i x_{ij}^B = \rho/m$, $j = 1, \ldots, m$;

(iii). x^*, the true optimal allocation computed by the method of Appendix 8.E;

as well as the $r_j^* = \sum_{i=1}^{n} \rho_i x_{ij}^*$ for the latter allocation. Table 8.1 contains the results for the case $c_i = c = 1$, $i = 1, \ldots, n$, like in Buzacott & Shanthikumar [64], i.e., $f(x)$ measures (twice) the overall mean waiting time for allocation x.

α	$f(x^S)$	$f(x^B)$	$f(x^*)$	r_1^*	r_2^*	r_3^*	r_4^*
0.01	0.05934	0.03747	0.03728	0.0876	0.0916	0.0986	0.0822
0.05	2.4545	1.5500	1.5468	0.4463	0.4502	0.4673	0.4362
0.10	54.100	34.100	34.086	0.8999	0.8980	0.9030	0.8992
0.11	653.40	412.61	412.46	0.9900	0.9897	0.9903	0.9900

TABLE 8.1. Value of objective function for x^B, x^S, and x^* with $c_j = c = 1$.

Table 8.1 supports the conclusion that in general the load should not be completely balanced, but suggests that for $c_i = c$, $i = 1, \ldots, n$, the quality of x^B tends to match that of x^*.

Table 8.2 contains the results for the case $c_i = \beta_i$, $i = 1, \ldots, n$, i.e., $f(x)$ measures (twice) the mean amount of waiting work for allocation x.

α	$f(x^S)$	$f(x^B)$	$f(x^*)$	r_1^*	r_2^*	r_3^*	r_4^*
0.01	0.11868	0.11868	0.10477	0.0553	0.0647	0.1140	0.1260
0.05	4.9091	4.9091	4.3542	0.3366	0.4105	0.5000	0.5528
0.10	108.00	108.00	94.941	0.8464	0.9024	0.9162	0.9351
0.11	1306.8	1306.8	1149.7	0.9840	0.9904	0.9917	0.9938

TABLE 8.2. Value of objective function for x^B, x^S, and x^* with $c_j = \beta_j$.

Table 8.2 suggests that for $c_i \neq c$, $i = 1, \ldots, n$, the quality of x^B in comparison with x^* tends to deteriorate.

8.5 FINDING AN OPTIMAL SOURCE PARTITIONING

In this section we consider the problem of finding an optimal *source partitioning*, i.e., a *deterministic* allocation of the customer types to the servers that minimizes the mean total waiting cost per unit of time. Using (8.1) and (8.4), we formulate the problem as follows.

Problem (II).

$$\text{minimize} \qquad f(x) = \sum_{j=1}^{m} \frac{\left(\sum_{i=1}^{n} \lambda_i c_i x_{ij}\right)\left(\sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x_{ij}\right)}{\mu_j\left(\mu_j - \sum_{i=1}^{n} \lambda_i \beta_i x_{ij}\right)} \tag{8.8}$$

$$\text{subject to} \qquad \sum_{i=1}^{n} \lambda_i \beta_i x_{ij} < \mu_j, \qquad j = 1, \ldots, m;$$

$$\sum_{j=1}^{m} x_{ij} = 1, \qquad i = 1, \ldots, n;$$

$$x_{ij} \in \{0, 1\}, \qquad i = 1, \ldots, n,\ j = 1, \ldots, m.$$

Problem (II) is a non-linear integer programming problem. Finding a feasible solution of problem (II) is equivalent to packing n elements of size $\rho_1, \ldots, \rho_n$ in m bins of capacity $\mu_1, \ldots, \mu_m$, which is known to be NP-hard for $m \geq 2$, even in the case $\mu_i = \mu/m$, cf. Garey & Johnson [105]. Finding an optimal solution is of course at least as hard as finding a feasible solution of problem (II). Concluding, even in the case of identical servers there is not likely to be an efficient way of solving problem (II), which motivates a heuristic approach. As even finding a feasible solution of problem (II) is NP-hard, we have to take for granted however that heuristic methods cannot be guaranteed to yield even a feasible solution. Globally speaking, the larger n and the larger $m-\rho$, the more feasible solutions there are and the more likely heuristic methods are to yield a feasible solution of problem (II). From a practical point of view, neither small n nor very small $m - \rho$ are of great concern with respect to solving problem (II). For small n the NP-hardness of problem (II) does not really matter, so that it be wiser to decide solving problem (II) by an enumerative method. For very small $m - \rho$ the system will be critically loaded anyway, so that it may be wiser in the design of the system to decide employing an extra server.

Below we indicate how the structure of an optimal solution of problem (I), the *non-integer* version (x_{ij}'s non-integer) of problem (II), may be used as a guideline in heuristically solving problem (II). To illustrate the specific complexity of problem (II), in comparison with problem (I), we first consider the special case of identical servers, i.e., $\mu_j = \mu/m$, $j = 1, \ldots, m$, and 'almost' identical customer types, i.e., $\left(\frac{c_i}{\beta_i}, \frac{\beta_i^{(2)}}{\beta_i}\right) = (\gamma, \delta)$, $i = 1, \ldots, n$.
Minimizing $f(x)$ then reduces to minimizing $g(x)$ defined as

$$g(x) := \sum_{j=1}^{m} \frac{\left(\sum_{i=1}^{n} \rho_i x_{ij}\right)^2}{\mu/m - \sum_{i=1}^{n} \rho_i x_{ij}}.$$

In a sense minimizing $g(x)$ amounts to making $\sum_{i=1}^{n} \rho_i x_{ij}$ for all $j = 1, \ldots, m$ as equal as possible. In particular, if $\sum_{i=1}^{n} \rho_i x_{ij} = \rho/m$ for all $j = 1, \ldots, m$, then $g(x)$ is certainly minimal, cf. Lemma 8.3.2. So an optimal solution of problem (I) is simply $x^*_{ij} = 1/m$, $i = 1, \ldots, n$, $j = 1, \ldots, m$. Due to the integrality conditions an optimal solution of problem (II) is however not so simple to obtain.

Minimizing $g(x)$ is strongly related to minimizing $h(x)$ defined as

$$h(x) := \max_{j=1,\ldots,m} \sum_{i=1}^{n} \rho_i x_{ij}.$$

Just like minimizing $g(x)$, minimizing $h(x)$ in a sense amounts to making $\sum_{i=1}^{n} \rho_i x_{ij}$ for all $j = 1, \ldots, m$ as equal as possible. In particular, if $\sum_{i=1}^{n} \rho_i x_{ij} = \rho/m$ for all $i = 1, \ldots, n$, then $h(x)$ is certainly minimal, again just like $g(x)$. For $m \leq 2$ minimizing $h(x)$ is completely equivalent to minimizing $g(x)$. In machine scheduling minimizing $h(x)$ is known as minimizing the makespan of n jobs of length $\rho_1, \ldots, \rho_n$ on m parallel identical machines, which is known to be NP-hard for $m \geq 2$, cf. Garey & Johnson [105]. (Of course this is not very surprising, knowing that packing is already NP-hard; minimizing the makespan is equivalent to packing the jobs in m bins of capacity as small as possible, which is of course at least as hard as packing the jobs in m bins of given capacity.)

A prominent family of heuristics for minimizing the makespan is the class of list scheduling rules. Characteristically of list scheduling rules, jobs are successively selected in order of appearance on some prespecified list, to be assigned to the machine with the least total processing time already assigned. The worst-case ratio for a list scheduling rule is $2 - 1/m$, cf. [110]. Evidently, the list ordering is the critical factor for the performance of a list scheduling rule. There is e.g. always a list ordering for which the list scheduling rule yields an optimal schedule. The worst-case instances suggest that the performance of a list scheduling rule may be better when jobs are selected in order of non-increasing ρ_i, which features the LPT (Longest Processing Time) rule. Indeed, the worst-case ratio for the LPT rule is $4/3 - 1/(3m)$, cf. [111]. Of course a worst-case ratio renders an inherently sombre picture, which not necessarily reflects the average performance. Probabilistic analysis reveals that the average performance of the LPT rule is indeed considerably better than the worst-case ratio may suggest, cf. [70].

Although minimizing $h(x)$ is closely related to minimizing $g(x)$, measures of the good performance with regard to $h(x)$ do not immediately carry over to measures of a good performance with regard to $g(x)$. The crux is that $g(x)$ is substantially more sensitive to suboptimality than $h(x)$; the larger the total traffic intensity, the more sensitive.

So far we considered the special case of identical servers and 'almost' identical customer types, in which it was easy to conclude that the total traffic stream should be balanced among the servers. Due to the integrality conditions it was however not so easy to accomplish an exactly balanced division. Returning now to the general case we cannot expect the picture to be brighter. On the contrary, in the general case it is not even known how exactly the total traffic stream should be divided among the servers, not to mention the problem of accomplishing a desirable division.
We now indicate how the structure of an optimal solution of problem (I) may be used as a guideline in heuristically solving problem (II). As the cost structure of problem (II) and problem (I) do not differ, there is no reason to believe that the structure of an optimal solution of problem (II) and problem (I) will dramatically differ. Globally speaking, the larger n and the larger $m-\rho$, the more feasible integer allocations there are and the more the structure of an optimal integer allocation is likely to resemble the structure of an optimal non-integer allocation. As remarked before, from a practical point of view, neither small n nor very small $m-\rho$ are of great concern with respect to solving problem (II). There are several options as to how the knowledge of an optimal solution of problem (I), the non-integer version of problem (II), may be used in heuristically solving problem (II).

Option 1.
A first option is to construct an integer allocation, starting from an optimal non-integer allocation. One may e.g. somehow round an optimal non-integer allocation x^* to an integer allocation x^{**}, taking into account the condition $\sum_{j=1}^{m} x_{ij}^{**} = 1$, $i = 1, \ldots, n$, i.e., $x_{ij}^{**} = 1$ for $i = i_j$, $x_{ij}^{**} = 0$ for $i \neq i_j$, for some i_j with $x_{i_j j}^* > 0$, $j = 1, \ldots, m$. As observed before, the optimality of clustering suggests that an optimal non-integer allocation is in fact 'almost' integer. Thus rounding may be expected to yield quite acceptable results. The main drawback is that an optimal non-integer allocation is needed, which is not so simple to obtain in cases with non-identical servers or where the customer types are not ordered.

Option 2.
A second option, which to some extent meets the drawback of the first option, is to construct an integer allocation, not starting from an optimal allocation *itself*, but from the *structure*. One may e.g. select the best from all integer allocations that satisfy Lemma 8.3.1, i.e., $P_{j'}(x^*) \cap P_{j''}(x^*) = \emptyset$ for $j' \neq j''$. It is easily verified that the best of all integer allocations that satisfy Lemma 8.3.1 is at least as good as the best integer allocation obtained from rounding an optimal non-integer allocation. The main drawback is that such a procedure, although polynomial in n for fixed m, may prove to be rather time-consuming.

Option 3.
A third option, which to some extent meets the drawback of the second option, is to construct an integer allocation, again starting from the structure of an optimal integer allocation, but not so rigorously. One may e.g. select the best from not all but some proper subclass of integer allocations that satisfy Lemma 8.3.1, i.e., integer allocations that satisfy $U_{j'}(x) \cap U_{j''}(x) = \emptyset$ for $j' \neq j''$ for some $U_j(x) \supseteq P_j(x)$, $j = 1, \ldots, m$. One may define e.g. $U_j(x) := \left[\min_{i \in K_j(x)} c_i/\beta_i, \max_{i \in K_j(x)} c_i/\beta_i\right] \times \left[\min_{i \in K_j(x)} \beta_i^{(2)}/\beta_i, \max_{i \in K_j(x)} \beta_i^{(2)}/\beta_i\right]$, thus blowing up the sets $P_j(x)$ to rectangles, so that indeed $P_j(x) \subseteq U_j(x)$, $j = 1, \ldots, m$.

8.6 Concluding remarks and suggestions for further research

We have considered the problem of finding an allocation that minimizes the mean total waiting cost per unit of time. We have shown that the customer types should be clustered according to the corresponding $\left(\frac{c_j}{\beta_j}, \frac{\beta_j^{(2)}}{\beta_j}\right)$-values and we described for some special cases in detail how that property may be exploited in computing an optimal allocation. In other cases (e.g. when the customer types are not ordered in the sense of Assumption 8.4.1 that property may still be exploited in calculating an optimal allocation, but not in a manner as simple. An interesting topic for further research might be to develop efficient (heuristic) methods for this. Further we have considered the problem of finding an optimal deterministic allocation. We have shown the problem to be NP-hard and indicated how the structure of an optimal non-deterministic allocation may be used as a heuristic guideline in searching for an optimal deterministic allocation. An interesting topic for further research might be to investigate the quality of the proposed heuristic methods, either in a theoretical framework or by numerical experiments. In either way, the quality of the heuristics may be judged by comparison with the optimal non-deterministic allocation which provides a lower bound for the optimal deterministic allocation.
In this chapter we have focused on the global scheduling problem; we have hardly touched on the local scheduling problem. We assumed the order of service not to discriminate between the various customer types. When the order of service *does* discriminate between the various customer types, the expressions for the mean waiting times are more complicated. However, for $c_i/\beta_i = \gamma$, $i = 1, \ldots, n$, for some γ, the results of this chapter still hold. Minimizing the mean total waiting cost per unit of time $\sum_{i=1}^{n} c_i \lambda_i \mathbf{E}\mathbf{W}_i$ then amounts to minimizing the mean amount of waiting work $\sum_{i=1}^{n} \rho_i \mathbf{E}\mathbf{W}_i$. Minimizing the mean amount of waiting work is equivalent to minimizing the mean total amount of work, as the difference, the mean amount of work in service, always equals $\frac{1}{2} \sum_{i=1}^{n} \lambda_i \beta_i^{(2)}$,

irrespective of the allocation x. A sample path comparison shows however that the total amount of work is not influenced by the local scheduling strategy, which property is frequently referred to as work conservation. Consequently minimizing $\sum_{i=1}^{n} \rho_i \mathbf{EW}_i$ is completely insensitive to the local scheduling strategy.
In this chapter we implicitly assumed the various customer types to be served without any interruptions. In some situations changing over from one customer type to another may however require a non-negligible switch-over time (e.g., tool changing in a flexible manufacturing system, traveling in the case of a repair crew visiting various installations). When the switch-over times are non-negligible, the expressions for the mean waiting times are more complicated. For some local scheduling strategies so-called pseudo-conservation laws provide comparatively simple expressions for $\sum_{i=1}^{n} \rho_i \mathbf{EW}_i$, but these expressions appear to be less amenable to optimization procedures than the objective function (8.4).
In this chapter we assumed that any customer type could, in principle, be allocated to any server. In some situations it may however occur that some customer types cannot be allocated to some servers, or that some customer types cannot be combined. Such restrictions may be translated into additional constraints on the routing probabilities, the x_{ij}'s. It would be interesting to investigate how those restrictions on the x_{ij}'s affect the structure of an optimal allocation.

Appendices

8.A Proof of Lemma 8.3.1

Lemma 8.3.1
$P_{j'}(x^*) \cap P_{j''}(x^*) = \emptyset$ for $j' \neq j''$.
In other words, if $\left(\frac{c_i}{\beta_i}, \frac{\beta_i^{(2)}}{\beta_i}\right) \in P_j(x^*)$, then $x_{ij}^* = 1$.

Proof
Suppose not, i.e., by definition of $P_j(x^*)$, there exist k_0, j', j'', $j' \neq j''$, such that

$$\left(\frac{c_{k_0}}{\beta_{k_0}}, \frac{\beta_{k_0}^{(2)}}{\beta_{k_0}}\right) = \sum_{k \in K_{j''}(x^*)} \alpha_k \left(\frac{c_k}{\beta_k}, \frac{\beta_k^{(2)}}{\beta_k}\right), \tag{8.9}$$

with $x_{k_0 j'}^* > 0$, $x_{k j''}^* > 0$, $\alpha_k \geq 0$, $k \in K_{j''}(x^*)$, $\sum_{k \in K_{j''}(x^*)} \alpha_k = 1$. Moreover, not all the points $\left(\frac{c_k}{\beta_k}, \frac{\beta_k^{(2)}}{\beta_k}\right)$ with $\alpha_k > 0$ lie on a line, i.e., there is no linear

equality that is satisfied by all the points $\left(\frac{c_k}{\beta_k}, \frac{\beta_k^{(2)}}{\beta_k}\right)$ with $\alpha_k > 0$.

Now consider $\epsilon \left\{ \frac{\partial f(x)}{\partial x_{k_0 j''}} - \frac{\partial f(x)}{\partial x_{k_0 j'}} \right\}_{|x=x^*}$ and $\epsilon \left\{ \frac{\partial f(x)}{\partial x_{kj'}} - \frac{\partial f(x)}{\partial x_{kj''}} \right\}_{|x=x^*}$, which measure the first-order effect on $f(x)$ around x^* of transferring a fraction ϵ of customer type k_0 from server j' to server j'' and transferring a fraction ϵ of customer type k from server j'' to server j', respectively, $k \in K_{j''}(x^*)$. As x^* is an optimal solution of problem (I), the first-order effect on $f(x)$ around x^* of transferring a customer type from one server to another cannot be negative, so $\left\{ \frac{\partial f(x)}{\partial x_{k_0 j''}} - \frac{\partial f(x)}{\partial x_{k_0 j'}} \right\}_{|x=x^*} \geq 0$, $\left\{ \frac{\partial f(x)}{\partial x_{kj'}} - \frac{\partial f(x)}{\partial x_{kj''}} \right\}_{|x=x^*} \geq 0$, $k \in K_{j''}(x^*)$, as may also be formally verified from the Kuhn-Tucker conditions.
Differentiating $f(\cdot)$ once,

$$\frac{\partial f(x)}{\partial x_{ij}} = \lambda_i c_i B_j(x) + \lambda_i \beta_i^{(2)} C_j(x) + \lambda_i \beta_i \mu_i B_j(x) C_j(x). \tag{8.10}$$

From (8.9),

$$\lambda_{k_0}(c_{k_0}, \beta_{k_0}^{(2)}, \beta_{k_0}) = \sum_{k \in K_{j''}(x^*)} \alpha_k \frac{\lambda_{k_0} \beta_{k_0}}{\lambda_k \beta_k} \lambda_k(c_k, \beta_k^{(2)}, \beta_k).$$

So

$$\left\{ \frac{\partial f(x)}{\partial x_{k_0 j''}} - \frac{\partial f(x)}{\partial x_{k_0 j'}} \right\}_{|x=x^*} = - \sum_{k \in K_{j''}(x^*)} \alpha_k \frac{\lambda_{k_0} \beta_{k_0}}{\lambda_k \beta_k} \left\{ \frac{\partial f(x)}{\partial x_{kj'}} - \frac{\partial f(x)}{\partial x_{kj''}} \right\}_{|x=x^*}.$$

As $\left\{ \frac{\partial f(x)}{\partial x_{k_0 j''}} - \frac{\partial f(x)}{\partial x_{k_0 j'}} \right\}_{|x=x^*} \geq 0$, $\alpha_k \left\{ \frac{\partial f(x)}{\partial x_{kj'}} - \frac{\partial f(x)}{\partial x_{kj''}} \right\}_{|x=x^*} \geq 0$, $k \in K_{j''}(x^*)$, we conclude that $\alpha_k \left\{ \frac{\partial f(x)}{\partial x_{kj'}} - \frac{\partial f(x)}{\partial x_{kj''}} \right\}_{|x=x^*} = 0$, $k \in K_{j''}(x^*)$. In other words,

$\frac{c_k}{\beta_k} (B_{j'}(x^*) - B_{j''}(x^*)) + \frac{\beta_k^{(2)}}{\beta_k} (C_{j'}(x^*) - C_{j''}(x^*)) = \mu_{j''} B_{j''}(x^*) C_{j''}(x^*) - \mu_{j'} B_{j'}(x^*) C_{j'}(x^*)$ for all the points $\left(\frac{c_k}{\beta_k}, \frac{\beta_k^{(2)}}{\beta_k}\right)$ with $\alpha_k > 0$, i.e., there *is* a linear equality that is satisfied by all the points $\left(\frac{c_k}{\beta_k}, \frac{\beta_k^{(2)}}{\beta_k}\right)$ with $\alpha_k > 0$. Contradiction.

□

8.B Proof of Lemma 8.3.2

Lemma 8.3.2
If $B_{j'}(x^*) \geq B_{j''}(x^*)$, $C_{j'}(x^*) \geq C_{j''}(x^*)$,
then $\mu_{j'} B_{j'}(x^*) C_{j'}(x^*) \leq \mu_{j''} B_{j''}(x^*) C_{j''}(x^*)$.

Proof
Suppose not, i.e.,

$$B_{j'}(x^*) \geq B_{j''}(x^*), C_{j'}(x^*) \geq C_{j''}(x^*), \tag{8.11}$$

$$\mu_{j'} B_{j'}(x^*) C_{j'}(x^*) > \mu_{j''} B_{j''}(x^*) C_{j''}(x^*).$$

Take k_0 such that $x^*_{k_0 j'} > 0$.
Now consider $\epsilon \left\{ \frac{\partial f(x)}{\partial x_{k_0 j''}} - \frac{\partial f(x)}{\partial x_{k_0 j'}} \right\}_{|x=x^*}$, which measures the first-order effect on $f(x)$ around x^* of transferring a fraction ϵ of customer type k_0 from server j' to server j''. As x^* is an optimal solution of problem (I), the first-order effect on $f(x)$ around x^* of transferring a customer type from one server to another cannot be negative, so $\left\{ \frac{\partial f(x)}{\partial x_{k_0 j''}} - \frac{\partial f(x)}{\partial x_{k_0 j'}} \right\}_{|x=x^*} \geq 0$, as may also be formally verified from the Kuhn-Tucker conditions.
From (8.10), if (8.11) were to hold,

$$\left\{ \frac{\partial f(x)}{\partial x_{k_0 j''}} - \frac{\partial f(x)}{\partial x_{k_0 j'}} \right\}_{|x=x^*} =$$

$$\lambda_{k_0} c_{k_0} \left(B_{j''}(x^*) - B_{j'}(x^*) \right) + \lambda_{k_0} \beta^{(2)}_{k_0} \left(C_{j''}(x^*) - C_{j'}(x^*) \right) +$$

$$\lambda_{k_0} \beta_{k_0} \left(\mu_{j''} B_{j''}(x^*) C_{j''}(x^*) - \mu_{j'} B_{j'}(x^*) C_{j'}(x^*) \right) < 0.$$

Contradiction. □

8.C Proof of Lemma 8.3.3

Lemma 8.3.3
Assume $x^*_{i'j'} > 0$, $x^*_{i''j''} > 0$.
If $\frac{c_{i'}}{\beta_{i'}} \leq \frac{c_{i''}}{\beta_{i''}}$, $\frac{\beta^{(2)}_{i'}}{\beta_{i'}} \geq \frac{\beta^{(2)}_{i''}}{\beta_{i''}}$, $\left(\frac{c_{i'}}{\beta_{i'}}, \frac{\beta^{(2)}_{i'}}{\beta_{i'}} \right) \neq \left(\frac{c_{i''}}{\beta_{i''}}, \frac{\beta^{(2)}_{i''}}{\beta_{i''}} \right)$, then $j' \leq j''$.

Proof
Suppose not, i.e., $x^*_{i'j'} > 0$, $x^*_{i''j''} > 0$, $j' \neq j''$,

$$\frac{c_{i'}}{\beta_{i'}} \leq \frac{c_{i''}}{\beta_{i''}}, \ \frac{\beta^{(2)}_{i'}}{\beta_{i'}} \geq \frac{\beta^{(2)}_{i''}}{\beta_{i''}}, \ \left(\frac{c_{i'}}{\beta_{i'}}, \frac{\beta^{(2)}_{i'}}{\beta_{i'}} \right) \neq \left(\frac{c_{i''}}{\beta_{i''}}, \frac{\beta^{(2)}_{i''}}{\beta_{i''}} \right), \tag{8.12}$$

$$B_{j'}(x^*) \leq B_{j''}(x^*),\ C_{j'}(x^*) \geq C_{j''}(x^*).$$

Now consider $\epsilon \left\{ \frac{\partial f(x)}{\partial x_{i'j''}} - \frac{\partial f(x)}{\partial x_{i'j'}} \right\}_{|x=x^*}$ and $\epsilon \left\{ \frac{\partial f(x)}{\partial x_{i''j'}} - \frac{\partial f(x)}{\partial x_{i''j''}} \right\}_{|x=x^*}$, which measure the first-order effect on $f(x)$ around x^* of transferring a fraction ϵ of customer type i' from server j' to server j'' and a fraction ϵ of customer type i'' from server j'' to server j', respectively. As x^* is an optimal solution of problem (I), the first-order effects on $f(x)$ around x^* of transferring customer types from one server to another cannot be negative, so $\left\{ \frac{\partial f(x)}{\partial x_{i'j''}} - \frac{\partial f(x)}{\partial x_{i'j'}} \right\}_{|x=x^*} \geq 0$, $\left\{ \frac{\partial f(x)}{\partial x_{i''j'}} - \frac{\partial f(x)}{\partial x_{i''j''}} \right\}_{|x=x^*} \geq 0$, as may also be formally verified from the Kuhn-Tucker conditions.
From (8.10),

$$\lambda_{i''}\beta_{i''} \left\{ \frac{\partial f(x)}{\partial x_{i'j''}} - \frac{\partial f(x)}{\partial x_{i'j'}} \right\} + \lambda_{i'}\beta_{i'} \left\{ \frac{\partial f(x)}{\partial x_{i''j'}} - \frac{\partial f(x)}{\partial x_{i''j''}} \right\} =$$

$$\lambda_{i'}\lambda_{i''}\beta_{i'}\beta_{i''} \left\{ \left(\frac{c_{i'}}{\beta_{i'}} - \frac{c_{i''}}{\beta_{i''}} \right) (B_{j''}(x) - B_{j'}(x)) + \right.$$

$$\left. \left(\frac{\beta_{i'}^{(2)}}{\beta_{i'}} - \frac{\beta_{i''}^{(2)}}{\beta_{i''}} \right) (C_{j''}(x) - C_{j'}(x)) \right\}.$$

So, if (8.12) were to hold, then

$$\lambda_{i''}\beta_{i''} \left\{ \frac{\partial f(x)}{\partial x_{i'j''}} - \frac{\partial f(x)}{\partial x_{i'j'}} \right\}_{|x=x^*} + \lambda_{i'}\beta_{i'} \left\{ \frac{\partial f(x)}{\partial x_{i''j'}} - \frac{\partial f(x)}{\partial x_{i''j''}} \right\}_{|x=x^*} \leq 0.$$

Contradiction, unless $\left\{ \frac{\partial f(x)}{\partial x_{i'j''}} - \frac{\partial f(x)}{\partial x_{i'j'}} \right\}_{|x=x^*} = 0$, $\left\{ \frac{\partial f(x)}{\partial x_{i''j'}} - \frac{\partial f(x)}{\partial x_{i''j''}} \right\}_{|x=x^*} = 0$.
Now consider $\epsilon^2 \sum_{i_1,i_2=i',i''} \sum_{j_1,j_2=j',j''} \left\{ \frac{\partial^2 f(x)}{\partial x_{i_1j_1}\partial x_{i_2j_2}} \alpha_{i_1j_1}\alpha_{i_2j_2} \right\}_{|x=x^*}$ for $\alpha_{i'j'} = -\lambda_{i''}\beta_{i''}$, $\alpha_{i'j''} = \lambda_{i''}\beta_{i''}$, $\alpha_{i''j'} = \lambda_{i'}\beta_{i'} + \alpha$, $\alpha_{i''j''} = -\lambda_{i'}\beta_{i'} - \alpha$, which measures the second-order effect on $f(x)$ around x^* of transferring a fraction $\epsilon\lambda_{i''}\beta_{i''}$ of customer type i' from server j' to server j'' and a fraction $\epsilon(\lambda_{i'}\beta_{i'} + \alpha)$ of customer type i'' from server j'' to server j'. As x^* is an optimal solution of problem (I), while the first-order effects on $f(x)$ around x^* of transferring customer types from one server to another are zero, the second-order effect cannot be negative, so $\sum_{i_1,i_2=i',i''} \sum_{j_1,j_2=j',j''} \left\{ \frac{\partial^2 f(x)}{\partial x_{i_1j_1}\partial x_{i_2j_2}} \alpha_{i_1j_1}\alpha_{i_2j_2} \right\}_{|x=x^*} \geq 0$.

Differentiating $f(\cdot)$ twice,

$$\frac{\partial^2 f(x)}{\partial x_{i_1 j_1} \partial x_{i_2 j_2}} = \delta_{j_1 j_2} \left\{ \frac{\lambda_{i_1}\lambda_{i_2}\left(c_{i_1}\beta_{i_2}^{(2)} + c_{i_2}\beta_{i_1}^{(2)}\right) + 2\lambda_{i_2}\beta_{i_2}\frac{\partial f(x)}{\partial x_{i_1 j_1}}}{\mu/m - \sum_{i=1}^{n}\lambda_i\beta_i x_{ij_1}} + \right.$$

$$\left. \frac{\lambda_{i_1}\lambda_{i_2}\left(\left(\beta_{i_1}c_{i_2} - \beta_{i_2}c_{i_1}\right)\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}x_{ij_1} + \left(\beta_{i_1}\beta_{i_2}^{(2)} - \beta_{i_2}\beta_{i_1}^{(2)}\right)\sum_{i=1}^{n}\lambda_i c_i x_{ij_1}\right)}{\left(\mu/m - \sum_{i=1}^{n}\lambda_i\beta_i x_{ij_1}\right)^2} \right\}$$

with $\delta_{j_1 j_2}$ the Kronecker delta. So, as $\frac{\partial f(x)}{\partial x_{i'j''}} - \frac{\partial f(x)}{\partial x_{i'j'}} = 0$, $\frac{\partial f(x)}{\partial x_{i''j'}} - \frac{\partial f(x)}{\partial x_{i''j''}} = 0$, $j' \neq j''$,

$$\sum_{i_1, i_2 = i', i''} \sum_{j_1, j_2 = j', j''} \frac{\partial^2 f(x)}{\partial x_{i_1 j_1}\partial x_{i_2 j_2}} \alpha_{i_1 j_1}\alpha_{i_2 j_2} =$$

$$2\left\{\frac{1}{\mu/m - \sum_{i=1}^{n}\lambda_i\beta_i x_{ij'}} + \frac{1}{\mu/m - \sum_{i=1}^{n}\lambda_i\beta_i x_{ij''}}\right\} \times$$

$$\left\{\lambda_{i'}^2\lambda_{i''}^2\beta_{i'}^2\beta_{i''}^2\left(\frac{c_{i''}}{\beta_{i''}} - \frac{c_{i'}}{\beta_{i'}}\right)\left(\frac{\beta_{i''}^{(2)}}{\beta_{i''}} - \frac{\beta_{i'}^{(2)}}{\beta_{i'}}\right) + \alpha\lambda_{i'}\lambda_{i''}^2\beta_{i'}\beta_{i''}^2 \times \right.$$

$$\left(\left(\frac{c_{i''}}{\beta_{i''}} - \frac{c_{i'}}{\beta_{i'}}\right)\left(\frac{\beta_{i''}^{(2)}}{\beta_{i''}} + B_{j'}(x)\right) + \left(\frac{\beta_{i''}^{(2)}}{\beta_{i''}} - \frac{\beta_{i'}^{(2)}}{\beta_{i'}}\right)\left(\frac{c_{i''}}{\beta_{i''}} + C_{j'}(x)\right)\right) +$$

$$\left.\alpha^2\left(\lambda_{i''}^2 c_{i''}\beta_{i''}^{(2)} + \lambda_{i''}\beta_{i''}\frac{\partial f(x)}{\partial x_{i''j'}}\right)\right\}.$$

So, if (8.12) were to hold, then $\sum_{i_1, i_2 = i', i''} \sum_{j_1, j_2 = j', j''} \left\{\frac{\partial^2 f(x)}{\partial x_{i_1 j_1}\partial x_{i_2 j_2}}\alpha_{i_1 j_1}\alpha_{i_2 j_2}\right\}_{|x=x^*} < 0$ for some α. Contradiction.

□

8.D Proof of Lemma 8.3.4

Lemma 8.3.4

$T_{j'}(x^*) \cap T_{j''}(x^*) = \emptyset$ for $j' \neq j''$.

In other words, if $\left(\frac{c_i}{\beta_i}, \frac{\beta_i^{(2)}}{\beta_i}\right) \in T_j(x^*)$, then $x_{ij}^* = 1$.

Proof

Suppose not, i.e., by definition of $T_j(x^*)$ there exist j', j'', $j' \neq j''$, $i'_{j'}, i''_{j'} \in K_{j'}(x^*)$, $i'_{j''}, i''_{j''} \in K_{j''}(x^*)$, y, z, such that

$$\frac{c_{i'_{j'}}}{\beta_{i'_{j'}}} \leq y \leq \frac{c_{i''_{j'}}}{\beta_{i''_{j'}}}, \frac{\beta^{(2)}_{i'_{j'}}}{\beta_{i'_{j'}}} \geq z \geq \frac{\beta^{(2)}_{i''_{j'}}}{\beta_{i''_{j'}}}, \left(\frac{c_{i'_{j'}}}{\beta_{i'_{j'}}}, \frac{\beta^{(2)}_{i'_{j'}}}{\beta_{i'_{j'}}}\right) \neq (y,z) \neq \left(\frac{c_{i''_{j'}}}{\beta_{i''_{j'}}}, \frac{\beta^{(2)}_{i''_{j'}}}{\beta_{i''_{j'}}}\right),$$

$$\frac{c_{i'_{j''}}}{\beta_{i'_{j''}}} \leq y \leq \frac{c_{i''_{j''}}}{\beta_{i''_{j''}}}, \frac{\beta^{(2)}_{i'_{j''}}}{\beta_{i'_{j''}}} \geq z \geq \frac{\beta^{(2)}_{i''_{j''}}}{\beta_{i''_{j''}}}, \left(\frac{c_{i'_{j''}}}{\beta_{i'_{j''}}}, \frac{\beta^{(2)}_{i'_{j''}}}{\beta_{i'_{j''}}}\right) \neq (y,z) \neq \left(\frac{c_{i''_{j''}}}{\beta_{i''_{j''}}}, \frac{\beta^{(2)}_{i''_{j''}}}{\beta_{i''_{j''}}}\right).$$

From Lemma 8.3.3, on the one hand, $j' \leq j''$, as $x^*_{i'_{j'}j'} > 0$, $x^*_{i''_{j''}j''} > 0$,

$$\frac{c_{i'_{j'}}}{\beta_{i'_{j'}}} \leq \frac{c_{i''_{j''}}}{\beta_{i''_{j''}}}, \frac{\beta^{(2)}_{i'_{j'}}}{\beta_{i'_{j'}}} \geq \frac{\beta^{(2)}_{i''_{j''}}}{\beta_{i''_{j''}}}, \left(\frac{c_{i'_{j'}}}{\beta_{i'_{j'}}}, \frac{\beta^{(2)}_{i'_{j'}}}{\beta_{i'_{j'}}}\right) \neq \left(\frac{c_{i''_{j''}}}{\beta_{i''_{j''}}}, \frac{\beta^{(2)}_{i''_{j''}}}{\beta_{i''_{j''}}}\right);$$

on the other hand, $i' \geq i''$, as $x^*_{i''_{j'}j'} > 0$, $x^*_{i''_{j''}j''} > 0$,

$$\frac{c_{i''_{j'}}}{\beta_{i''_{j'}}} \geq \frac{c_{i'_{j''}}}{\beta_{i'_{j''}}}, \frac{\beta^{(2)}_{i''_{j'}}}{\beta_{i''_{j'}}} \leq \frac{\beta^{(2)}_{i'_{j''}}}{\beta_{i'_{j''}}}, \left(\frac{c_{i''_{j'}}}{\beta_{i''_{j'}}}, \frac{\beta^{(2)}_{i''_{j'}}}{\beta_{i''_{j'}}}\right) \neq \left(\frac{c_{i'_{j''}}}{\beta_{i'_{j''}}}, \frac{\beta^{(2)}_{i'_{j''}}}{\beta_{i'_{j''}}}\right).$$

So $j' = j''$. Contradiction.

□

8.E A METHOD FOR DETERMINING AN OPTIMAL ALLOCATION

In this appendix we describe a method for determining an optimal allocation in cases that satisfy Assumption 8.4.1. In Section 8.4 we sketched the main idea of the method. Here we describe the method in detail.

Lemma 8.E.1

a. $\displaystyle\sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x^*_{i1} \geq \frac{1}{m} \sum_{i=1}^{n} \lambda_i \beta_i^{(2)} \geq \sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x^*_{im}.$

b. $\displaystyle\sum_{i=1}^{n} \lambda_i c_i x^*_{i1} \leq \frac{1}{m} \sum_{i=1}^{n} \lambda_i c_i \leq \sum_{i=1}^{n} \lambda_i c_i x^*_{im}.$

Proof

Proof of a. The servers are indexed such that

$$\frac{\displaystyle\sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x^*_{ij}}{\mu/m - \displaystyle\sum_{i=1}^{n} \lambda_i \beta_i x^*_{ij}} \geq \frac{\displaystyle\sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x^*_{ij+1}}{\mu/m - \displaystyle\sum_{i=1}^{n} \lambda_i \beta_i x^*_{ij+1}}.$$

So

$$\frac{\mu}{m}\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}x_{ij}^{*}-\left(\sum_{i=1}^{n}\lambda_i\beta_i x_{ij+1}^{*}\right)\left(\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}x_{ij}^{*}\right)\geq \tag{8.13}$$

$$\frac{\mu}{m}\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}x_{ij+1}^{*}-\left(\sum_{i=1}^{n}\lambda_i\beta_i x_{ij}^{*}\right)\left(\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}x_{ij+1}^{*}\right).$$

On the one hand

$$\frac{\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}x_{ij}^{*}}{\sum_{i=1}^{n}\lambda_i\beta_i x_{ij}^{*}}\geq\min_{i\in K_j(x^*)}\frac{\beta_i^{(2)}}{\beta_i};$$

on the other hand

$$\frac{\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}x_{ij+1}^{*}}{\sum_{i=1}^{n}\lambda_i\beta_i x_{ij+1}^{*}}\leq\max_{i\in K_{j+1}(x^*)}\frac{\beta_i^{(2)}}{\beta_i}.$$

From Lemma 8.3.3, $\min_{i\in K_j(x^*)}\frac{\beta_i^{(2)}}{\beta_i}\geq\max_{i\in K_{j+1}(x^*)}\frac{\beta_i^{(2)}}{\beta_i}$.

So

$$\left(\sum_{i=1}^{n}\lambda_i\beta_i x_{ij+1}^{*}\right)\left(\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}x_{ij}^{*}\right)\geq \tag{8.14}$$

$$\left(\sum_{i=1}^{n}\lambda_i\beta_i x_{ij}^{*}\right)\left(\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}x_{ij+1}^{*}\right).$$

Combining (8.13), (8.14), $\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}x_{ij}^{*}\geq\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}x_{ij+1}^{*}$. Further $\sum_{j=1}^{m}\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}x_{ij}^{*}$ $=\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}$, as $\sum_{j=1}^{m}x_{ij}^{*}=1$, $i=1,\ldots,n$. So $\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}x_{i1}^{*}\geq\frac{1}{m}\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}\geq$ $\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}x_{im}^{*}$.

Proof of b. Follows immediately by symmetry considerations.

□

Step 1. Start with the allocation to server 1, which is to carry the largest $B_j(x^*)$ and the smallest $C_j(x^*)$.

Determine a lower bound $s_1^l = \sum_{i=1}^{n} x_{i1}^l$ for s_1^* from (i) $\sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x_{i1}^l = \frac{1}{m} \sum_{i=1}^{n} \lambda_i \beta_i^{(2)}$, (ii) if $x_{i'1}^l > 0$, then $x_{i''1}^l = 1$ for $i'' < i'$. However, if $\sum_{i=1}^{n} \lambda_i \beta_i^{(2)} x_{i1}^l = \frac{1}{m} \sum_{i=1}^{n} \lambda_i \beta_i^{(2)}$ implies $\sum_{i=1}^{n} \lambda_i \beta_i x_{i1}^l \geq 1$, then replace (i) by (iii) $\sum_{i=1}^{n} \lambda_i \beta_i x_{i1}^l = 1 - \epsilon$, with ϵ sufficiently small.

Determine an upper bound $s_1^u = \sum_{i=1}^{n} x_{i1}^u$ for s_1^* from (i) $\sum_{i=1}^{n} \lambda_i c_i x_{i1}^u = \frac{1}{m} \sum_{i=1}^{n} \lambda_i c_i$, (ii) if $x_{i'1}^u > 0$, then $x_{i''1}^u = 1$ for $i'' < i'$. However, if $\sum_{i=1}^{n} \lambda_i c_i x_{i1}^u = \frac{1}{m} \sum_{i=1}^{n} \lambda_i c_i$ implies $\sum_{i=1}^{n} \lambda_i \beta_i x_{i1}^u \leq \rho - (m-1)$, then replace (i) by (iii) $\sum_{i=1}^{n} \lambda_i \beta_i x_{i1}^u = \rho - (m-1) + \epsilon$, with ϵ sufficiently small.

Step 2. Make an estimate $s_1 = \sum_{i=1}^{n} x_{i1}$ for s_1^*, somewhere in between s_1^l and s_1^u, e.g., $s_1 = (s_1^l + s_1^u)/2$.

Step 3. For $j = 1, 2, \ldots, m-1$, determine $s_{j+1} = \sum_{i=1}^{n} x_{ij+1}$ from (i) $x_{i_j j+1} \leq 1 - \sum_{k=1}^{j} x_{i_j k}$, with $i_j = \max\{i \mid x_{ij} > 0\}$, (ii) if $x_{i'j+1} > 0$ for $i_j < i'$, then $x_{i_j j+1} = 1 - \sum_{k=1}^{j} x_{i_j k}$, $x_{i''j+1} = 1$ for $i_j < i'' < i'$, (iii) $\frac{\partial f(x)}{\partial x_{i_j j}} = \frac{\partial f(x)}{\partial x_{i_j j+1}}$. Provided $i_j = \max\{i \mid x_{ij} > 0\}$, the latter condition necessarily holds for an optimal solution of problem (I), as may be formally verified from the Kuhn-Tucker conditions. Note that $\frac{\partial f(x)}{\partial x_{i_j j}} > 0$ and constant in s_{j+1}, and $\frac{\partial f(x)}{\partial x_{i_j j+1}} = 0$ for $s_{j+1} = 0$ and increasing in s_{j+1}. So through (i), (ii), (iii), s_j uniquely determines s_{j+1}, unless $\frac{\partial f(x)}{\partial x_{i_j j}} > \frac{\partial f(x)}{\partial x_{i_j j+1}}$ even for $x_{nj+1} = 1$, to which we return in the next step.

Step 4. Sooner or later one either runs out of servers, i.e., $\frac{\partial f(x)}{\partial x_{i_{m-1} m-1}} = \frac{\partial f(x)}{\partial x_{i_{m-1} m}}$, but $x_{nm} < 1$, or out of customer types, i.e., $x_{nj+1} = 1$, but $\frac{\partial f(x)}{\partial x_{i_j j}} > \frac{\partial f(x)}{\partial x_{i_j j+1}}$. If one runs out of servers, then apparently $s_1 < s_1^*$, so then replace s_1^l by s_1. If one runs out of customer types, then apparently $s_1 > s_1^*$, so then replace s_1^u by s_1. One may of course make a more sophisticated estimate for s_1^* than the estimate suggested above. One may e.g. use information about how soon one either ran out of servers or out of customer types. Repeat the procedure until lower and upper bound are sufficiently close.

Chapter 9

Polling systems with multiple coupled servers

9.1 Introduction

In the present chapter we consider polling systems with multiple coupled servers. So far there are hardly any exact results known for multiple-server polling systems, apart from some mean value results for global performance measures like cycle times. In this chapter we explore the class of systems that allow an exact analysis.

Polling systems with multiple servers have received remarkably little attention in the vast literature on polling systems. One of the first studies is Morris & Wang [153] in which the servers are assumed to be independent, i.e., the servers visit the queues independently of each other, each server according to some cyclic schedule. A very interesting phenomenon observed by Morris & Wang is the tendency for the servers to cluster if they follow identical routes, especially in heavy traffic. The phenomenon may be visualized as follows. A trailing server will tend to move fast, as it only encounters recently served queues, whereas a leading server will tend to be slowed down by queues that have not been served for a while, so that the servers tend to form bunches while constantly leapfrogging over one another.

Browne & Weiss [58] is one of the few studies in which the servers are assumed to be coupled, i.e., the servers always visit the queues together. They obtain index-type rules for determining the visit order that minimizes the mean length of individual cycles for both the exhaustive and the gated service discipline. Browne et al. [56] derive the mean waiting time for a completely symmetric two-queue system with an infinite number of coupled servers and deterministic service times. Browne & Kella [57] obtain the busy-period distribution for a two-queue system with an infinite number of coupled servers, exhaustive ser-

vice, and deterministic service times at one queue and general service times at the other queue.

Levy & Yechiali [143] and Kao & Narayanan [122] study the joint distribution of the queue length and the number of busy servers for a Markovian multiple-server queue, where the servers individually go on vacation when there are no waiting customers left. Mitrany & Avi-Itzhak [152] and Neuts & Lucantoni [155] analyze the joint distribution of the queue length and the number of busy servers for a Markovian multiple-server queue where servers break down at exponential intervals and then get repaired. For references on an approximative analysis of models with multiple queues and models with independent servers we refer to the introduction of the next chapter.

All these studies unanimously point out that multiple-server polling systems, combining the complexity of single-server polling systems and multiple-server systems, are extraordinarily hard to analyze. In fact, none of the studies (except [56], [57] for very specific two-queue infinite-server cases) presents any exact results for systems with multiple queues, apart from some mean-value results for global performance measures like cycle times.

In this chapter we consider the case of coupled servers. We are mainly interested in exploring the class of systems that allow an exact analysis. For these systems we present distributional results for the waiting time, the marginal queue length, and the joint queue length at polling epochs. The motivation for considering the case of coupled servers is threefold. First, the dependence in the position of the servers does not play any complicating role then. Second, in some situations the servers may in fact happen to be physically coupled. Third, the coupled server case may also be relevant for the study of the independent server case. The tendency for the servers to cluster provides e.g. an indication that they tend to behave as if they were coupled.

The remainder of the chapter is organized as follows. In Section 9.2 we consider a single-queue multiple-server system with service interruptions, which is not only interesting in its own right but also useful for the study of a multiple-server polling system. As described in Section 2.1, in isolation a specific queue in a polling system may be viewed as a single-queue system with service interruptions, the intervisit periods constituting the service interruptions. Results for single-queue systems with service interruptions may thus be used to obtain results for the marginal distributions in polling systems. We then return to the multiple-server polling system. In Section 9.3 we present a detailed model description. In Section 9.4 we relate the probability generating function (pgf) of the joint queue length at the beginning of a visit to the pgf of the joint queue length at the end of the *previous* visit. Next we relate the pgf of the joint queue length distribution at the *end* of a visit to the pgf of the joint queue length at the *beginning* of a visit. Thus we obtain $2n$ equations involving $2n$ pgf's, with n the number of queues. In Section 9.5 we identify some cases for which these pgf's can actually be solved from these equations. These cases include several single-queue systems with a varying number of servers, two-queue two-server systems with exhaustive service and exponential service times, as well

as infinite-server systems with an arbitrary number of queues, exhaustive or gated service, and deterministic service times. In Section 9.6 we conclude with some remarks and suggestions for further research.

9.2 An $M/M/m$ queue with coupled servers and service interruptions

In this section we consider an $M/M/m$ queue with coupled servers and service interruptions. The service interruptions are assumed to result from some interfering process that from time to time may keep the servers from working, even when there are customers present. Service preemption due to service interruptions is allowed. The service interruptions may be interwoven with the arrival and service processes in an arbitrarily complex manner, but may not anticipate on the arrival and service times of future customers. In particular the *durations* of successive service interruptions are allowed to be dependent. We abstract here from what kind of interfering process causes the service interruptions. In the context of polling models a service interruption typically models the intervisit period. In the setting of performability models a service interruption usually represents a down-period of the system. A period during which none of the servers is busy, because of a service interruption or because there are no customers present, will be called a non-serving interval. A period during which at least one of the servers is busy, will be called a serving interval.

Fuhrmann & Cooper [102] consider an $M/G/1$ queue with service interruptions. Under rather mild assumptions they prove a decomposition property of the queue length distribution. Using concepts from the theory of branching processes they show that the queue length distribution can be expressed as the convolution of the distribution of the following two quantities:

(i). the queue length at an arbitrary epoch in the 'corresponding' $M/G/1$ queue without service interruptions;

(ii). the queue length at an arbitrary epoch in a non-serving interval.

The 'corresponding' $M/G/1$ queue without service interruptions is an ordinary $M/G/1$ queue with similar traffic characteristics, of which the queue length distribution is simply known from the Pollaczek-Khintchine formula. To find the queue length distribution at an arbitrary epoch, it thus suffices to find the queue length distribution in a non-serving interval, which is quite often relatively simple. By the distributional form of Little's law the queue length decomposition also translates into a decomposition of the waiting time. Under somewhat milder assumptions than Fuhrmann & Cooper, Boxma [38] proves a similar decomposition of the amount of work in the system. Browne & Kella [57] analyze the queue length distribution in an $M/G/\infty$ queue with vacations. They observe that for deterministic service times a Fuhrmann & Cooper-like decomposition property holds but not for exponential service times.

We now analyze the queue length distribution in the $M/M/m$ queue under consideration. Although for $m = 1$ the amount of work is somewhat easier to

study than the queue length, for $m > 1$ we need to focus on the queue length, as the amount of work then no longer completely determines the number of busy servers. We make the following assumptions.
(i). During a serving interval there are no servers idling while there are customers waiting, i.e., if there are l customers present during a serving interval then there are $\min(l, m)$ servers working, just like in an ordinary $M/M/m$ queue.
(ii). The order in which customers enter service is independent of their service times.
Under the above assumptions we will show that the queue length distribution can be expressed into the distribution of (conceptually) the same two quantities as in the $M/G/1$ queue with service interruptions, but not in the same simple convolution form. However, to find the queue length distribution at an arbitrary epoch, it still suffices to find the queue length distribution in a non-serving interval. Under some additional assumptions we will also show how the queue length decomposition translates into a decomposition of the waiting time.
We first introduce some notation. Let λ be the arrival rate and let μ be the service rate. Define $\rho := \lambda/\mu$. Denote by $\mathbf{N}$ and $\mathbf{N}_I$ the total number of customers present (including customers in service) at, respectively, an arbitrary epoch and an arbitrary epoch in a non-serving interval. Denote by $\mathbf{N}^{(l)}_{M/M/m}$ the number of customers at an arbitrary epoch in the 'corresponding' $M/M/m$ queue, given that the number of customers is at least l, $l \geq 0$. The 'corresponding' $M/M/m$ queue is an ordinary $M/M/m$ queue with arrival rate λ and service rate μ.
For $l \leq m-1$,

$$\mathrm{E}(z^{\mathbf{N}^{(l)}_{M/M/m}}) = \tag{9.1}$$

$$\left[\sum_{k=l}^{m-1} \frac{\rho^k}{k!} + \frac{\rho^m}{m!}\frac{m}{m-\rho}\right]^{-1} \left[\sum_{k=l}^{m-1} z^k \frac{\rho^k}{k!} + z^m \frac{\rho^m}{m!}\frac{m}{m-\rho z}\right].$$

For $l \geq m-1$,

$$\mathrm{E}(z^{\mathbf{N}^{(l)}_{M/M/m}}) = z^l \frac{m-\rho}{m-\rho z}. \tag{9.2}$$

Lemma 9.2.1 expresses the distribution of $\mathbf{N}$ into the distribution of $\mathbf{N}_I$ and $\mathbf{N}^{(l)}_{M/M/m}$.

Lemma 9.2.1

$$\mathrm{E}(z^{\mathbf{N}}) = \tag{9.3}$$

$$\gamma\left[\sum_{l=0}^{m-2} \frac{\mathrm{E}(z^{\mathbf{N}^{(l)}_{M/M/m}})\Pr\{\mathbf{N}_I = l\}}{\Pr\{\mathbf{N}^{(l)}_{M/M/m} = l\}} + \frac{m}{m-\rho z}\sum_{l=m-1}^{\infty} z^l \Pr\{\mathbf{N}_I = l\}\right],$$

with

$$\gamma = \left[\sum_{l=0}^{m-2} \frac{\Pr\{\mathbf{N}_I = l\}}{\Pr\{\mathbf{N}^{(l)}_{M/M/m} = l\}} + \frac{m}{m-\rho} \sum_{l=m-1}^{\infty} \Pr\{\mathbf{N}_I = l\} \right]^{-1}. \tag{9.4}$$

Proof
See Appendix 9.A.

□

Remark 9.2.1
For $\Pr\{\mathbf{N}_I = 0\} = 1$, i.e., in a non-serving interval there are never any customers present, (9.3) and (9.4) reduce to

$$\mathrm{E}(z^{\mathbf{N}}) = \left[\sum_{l=0}^{m-1} \frac{\rho^l}{l!} + \frac{\rho^m}{m!} \frac{m}{m-\rho} \right]^{-1} \left[\sum_{l=0}^{m-1} z^l \frac{\rho^l}{l!} + z^m \frac{\rho^m}{m!} \frac{m}{m-\rho z} \right],$$

which is of course just the queue length distribution at an arbitrary epoch in the corresponding $M/M/m$ queue without service interruptions.

□

Remark 9.2.2
For $m = 1$, (9.3) and (9.4) reduce to

$$\mathrm{E}(z^{\mathbf{N}}) = \frac{1-\rho}{1-\rho z} \mathrm{E}(z^{\mathbf{N}_I}),$$

which is the Fuhrmann-Cooper decomposition for an $M/M/1$ queue with service interruptions.
For $m = \infty$, (9.3) and (9.4) reduce to

$$\mathrm{E}(z^{\mathbf{N}}) = \left[\sum_{l=0}^{\infty} \Pr\{\mathbf{N}_I = l\} \frac{l!}{\rho^l} \sum_{k=l}^{\infty} \frac{\rho^k}{k!} \right]^{-1} \left[\sum_{l=0}^{\infty} \Pr\{\mathbf{N}_I = l\} \frac{l!}{\rho^l} \sum_{k=l}^{\infty} z^k \frac{\rho^k}{k!} \right].$$

Recognizing that $\dfrac{l!}{\rho^l} \displaystyle\sum_{k=l}^{\infty} z^k \frac{\rho^k}{k!} = z^l + \rho \int_{u=0}^{z} u^l \mathrm{e}^{(z-u)\rho} du,$

$$\mathrm{E}(z^{\mathbf{N}}) = \tag{9.5}$$

$$\left[1 + \rho \int_{u=0}^{1} u^l \mathrm{e}^{(1-u)\rho} du \right]^{-1} \left[\mathrm{E}(z^{\mathbf{N}_I}) + \rho \int_{u=0}^{z} \mathrm{E}(u^{\mathbf{N}_I}) \mathrm{e}^{(z-u)\rho} du \right],$$

which will be useful later on.

□

Lemma 9.2.1 implies that to find the distribution of $\mathbf{N}$, it suffices to find the distribution of $\mathbf{N}_I$, as the distribution of $\mathbf{N}^{(l)}_{M/M/m}$ is known from (9.1). From a methodological point of view however it is more natural to analyze the queue length at either the beginning or the end of non-serving intervals than to study $\mathbf{N}_I$, the queue length at an arbitrary epoch in a non-serving interval. Therefore we now relate the distribution of $\mathbf{N}_I$ to the queue length distribution at such embedded epochs. Denote by $\mathbf{N}^{(k)}_{\text{begin}}$ and $\mathbf{N}^{(k)}_{\text{end}}$ the queue length at, respectively, the beginning and the end of the k-th non-serving interval. Denote by $\mathbf{N}_{\text{begin}}$, $\mathbf{N}_{\text{end}}$ a pair of stochastic variables with as joint distribution the stationary joint distribution of $\mathbf{N}^{(k)}_{\text{begin}}$, $\mathbf{N}^{(k)}_{\text{end}}$.

Lemma 9.2.2 relates the distribution of $\mathbf{N}_I$ to the distribution of $\mathbf{N}_{\text{begin}}$ and $\mathbf{N}_{\text{end}}$.

Lemma 9.2.2

$$\Pr\{\mathbf{N}_I = l\} = \frac{\Pr\{\mathbf{N}_{\text{begin}} \leq l\} - \Pr\{\mathbf{N}_{\text{end}} \leq l\}}{\mathrm{E}\mathbf{N}_{\text{end}} - \mathrm{E}\mathbf{N}_{\text{begin}}}. \tag{9.6}$$

Written in terms of pgf's

$$\mathrm{E}(z^{\mathbf{N}_I}) = \frac{\mathrm{E}(z^{\mathbf{N}_{\text{begin}}}) - \mathrm{E}(z^{\mathbf{N}_{\text{end}}})}{(1-z)(\mathrm{E}\mathbf{N}_{\text{end}} - \mathrm{E}\mathbf{N}_{\text{begin}})}. \tag{9.7}$$

Proof

The proof is completely similar to the proof of Lemma 2.2.1. Note that the proof of Lemma 2.2.1 does not rely on the assumption of a single server as implicitly made in Chapter 2.

□

To illustrate Lemma 9.2.2 we now give two simple examples.

Example 9.2.1

Consider an exhaustive vacation system, where the servers together go on vacation when the system is empty. Then $\mathrm{E}(z^{\mathbf{N}_{\text{begin}}}) = 1$. Let $\mathbf{V}$ be the length of an arbitrary vacation. Let $v(\omega) = \mathrm{E}(\mathrm{e}^{-\omega \mathbf{V}})$ for $\mathrm{Re}\,\omega \geq 0$. If the system is still empty when the servers return from vacation, then in the multiple-vacation case the servers again go on vacation, i.e., $\mathrm{E}(z^{\mathbf{N}_{\text{end}}}) = \dfrac{v(\lambda(1-z)) - v(\lambda)}{1 - v(\lambda)}$, whereas in the single-vacation case the servers just remain idling, awaiting a customer to arrive, i.e., $\mathrm{E}(z^{\mathbf{N}_{\text{end}}}) = v(\lambda(1-z)) - v(\lambda)(1-z)$. Summarizing, in the multiple-vacation case

$$\mathrm{E}(z^{\mathbf{N}_I}) = \frac{1 - v(\lambda(1-z))}{(1-z)\lambda \mathrm{E}\mathbf{V}}, \tag{9.8}$$

whereas in the single-vacation case

$$\mathrm{E}(z^{\mathbf{N}_I}) = \frac{1 - v(\lambda(1-z)) + v(\lambda)(1-z)}{(1-z)(\lambda \mathrm{E}\mathbf{V} + v(\lambda))}. \tag{9.9}$$

Browne & Kella [57] analyze an exhaustive $M/M/\infty$ system with multiple vacations. By a direct method they find $\mathrm{E}(z^{\mathbf{N}}) = H_{\mathbf{V}}(z)/H_{\mathbf{V}}(1)$ with

$$H_{\mathbf{V}}(z) = \frac{1 - v(\lambda(1-z))}{(1-z)\lambda \mathrm{E}\mathbf{V}} + \rho \int_{u=0}^{z} \frac{1 - v(\lambda(1-u))}{(1-u)\lambda \mathrm{E}\mathbf{V}} \mathrm{e}^{(z-u)\rho} du,$$

which agrees with (9.5), (9.8).

□

Example 9.2.2

Consider an exhaustive system with a stochastic $\mathbf{K}$-policy, cf. Bisdikian [19], where service is interrupted when the system is empty, while service is resumed as soon as a stochastic number of $\mathbf{K}$ customers have accumulated again. Then

$$\mathrm{E}(z^{\mathbf{N}_I}) = \frac{1 - \mathrm{E}(z^{\mathbf{K}})}{(1-z)\mathrm{E}\mathbf{K}}. \tag{9.10}$$

Browne & Kella [57] also study an exhaustive $M/M/\infty$ system with a deterministic K-policy. By a direct method they find $\mathrm{E}(z^{\mathbf{N}}) = H_K(z)/H_K(1)$ with

$$H_K(z) = \frac{1 - z^K}{1-z} + \rho \sum_{k=0}^{K-1} \int_{u=0}^{z} u^k \mathrm{e}^{(z-u)\rho} du,$$

which agrees with (9.5), (9.10).

□

We now show how the queue length decomposition translates into a decomposition of the waiting time. In addition to (i) and (ii) we make the following assumptions.

(iii). Customers enter service in order of arrival.

(iv). The waiting time of customers is independent of arrivals after their own arrival.

Denote by $\mathbf{W}$ and $\mathbf{R}$ respectively the waiting and the sojourn time of an arbitrary customer. Denote by $\mathbf{L}$ the number of waiting customers at an arbitrary epoch. The familiar relationship $\mathrm{E}(z^{\mathbf{N}}) = \mathrm{E}(\mathrm{e}^{-\lambda(1-z)\mathbf{R}})$ does *not* hold here, as customers do not necessarily leave in order of arrival. However, what *does* hold under the assumptions (iii) and (iv) is the relationship $\mathrm{E}(z^{\mathbf{L}}) = \mathrm{E}(\mathrm{e}^{-\lambda(1-z)\mathbf{W}})$. What thus remains to be done, is to relate the distribution of $\mathbf{L}$ to the distribution of $\mathbf{N}$.

Lemma 9.2.3

$$\mathrm{E}(z^{\mathbf{L}}) = \tag{9.11}$$
$$\mathrm{E}(z^{\mathbf{N}}) + (1-p_I)\left([z^{-m}-1]\mathrm{E}(z^{\mathbf{N}_E}) + \sum_{k=0}^{m-1}[1-z^{k-m}]\Pr\{\mathbf{N}_E = k\}\right),$$

with

$$\mathrm{E}(z^{\mathbf{N}_E}) = \frac{\mathrm{E}(z^{\mathbf{N}}) - p_I\mathrm{E}(z^{\mathbf{N}_I})}{1-p_I}, \tag{9.12}$$

$$p_I = \left[\sum_{l=0}^{\infty}\frac{\Pr\{\mathbf{N}_I = l\}}{\Pr\{\mathbf{N}^{(l)}_{M/M/m} = l\}}\right]^{-1}.$$

Proof

Denote by $\mathbf{N}_E$ the number of customers present at an arbitrary epoch in a serving interval. Denote by p_I the fraction of time occupied by non-serving intervals. Then

$$\mathrm{E}(z^{\mathbf{N}}) = \mathrm{E}(z^{\mathbf{N}_I})p_I + \mathrm{E}(z^{\mathbf{N}_E})(1-p_I), \tag{9.13}$$

$$\mathrm{E}(z^{\mathbf{L}}) = \mathrm{E}(z^{\mathbf{N}_I})p_I + \mathrm{E}(z^{[\mathbf{N}_E-m]^+})(1-p_I), \tag{9.14}$$

with $[x]^+ = \max(0,x)$.
Comparing (9.13), (9.14), using that

$$\mathrm{E}(z^{[\mathbf{N}_E-m]^+}) = z^{-m}\mathrm{E}(z^{\mathbf{N}_E}) + \sum_{k=0}^{m-1}[1-z^{k-m}]\Pr\{\mathbf{N}_E = k\} \tag{9.15}$$

yields (9.11).
Because of the PASTA property p_I equals the probability that an arbitrary customer arrives in a non-serving interval. In the proof of Lemma 9.2.1 we introduced the notion of a fundamental period. We showed that in a fundamental period exactly 1 customer is served that arrived in a non-serving interval. Denote by $\mathbf{M}$ the number of customers served in a fundamental period. Then

$$p_I = \frac{1}{\mathrm{EM}}. \tag{9.16}$$

From the proof of Lemma 9.2.1

$$\mathrm{EM} = \sum_{l=0}^{\infty}\frac{\Pr\{\mathbf{N}_I = l\}}{\Pr\{\mathbf{N}^{(l)}_{M/M/m} = l\}}. \tag{9.17}$$

Substituting (9.17) into (9.16) completes the proof.

□

9.3 MODEL DESCRIPTION

We now return to the polling system with multiple coupled servers. We first present a detailed model description. The model under consideration consists of n queues, $Q_1, \ldots, Q_n$, each of infinite capacity, attended by m coupled servers. For the specification of the arrival, service, and switch-over processes we refer to the description of the 'basic model' in Section 1.3.
The server pool visits the queues in a strictly cyclic order, $Q_1, \ldots, Q_n$. As soon as the servers arrive at Q_i, they start serving type-i customers, as prescribed by the service discipline. For now we do not specify the service discipline any further. In fact, what we are mainly interested in, is exploring the class of service disciplines that allow an exact analysis. As soon as the servers have finished serving type-i customers, as prescribed by the service discipline, they move to Q_{i+1}.

9.4 THE JOINT QUEUE LENGTH DISTRIBUTION I

In this section we relate the pgf of the joint queue length distribution at the beginning of a visit to Q_i to the pgf of the joint queue length distribution at the end of a visit to Q_{i-1}. Next we also relate the pgf of the joint queue length distribution at the *end* of a visit to Q_i to the pgf of the joint queue length distribution at the *beginning* of a visit to Q_i. Thus we obtain $2n$ equations involving $2n$ pgf's. In the next section we identify some cases in which these pgf's can actually be solved from these equations.
We first introduce some notation. Denote by $\mathbf{X}_{ih}$ and $\mathbf{Y}_{ih}$ stochastic variables with distribution the stationary queue length distribution at Q_h at, respectively, the beginning and the end of a visit to Q_i, $h = 1, \ldots, n$, $i = 1, \ldots, n$.
Define

$$F_i(z) := \mathrm{E}(z_1^{\mathbf{X}_{i1}} \ldots z_n^{\mathbf{X}_{in}})$$

$$G_i(z) := \mathrm{E}(z_1^{\mathbf{Y}_{i1}} \ldots z_n^{\mathbf{Y}_{in}})$$

for $z = (z_1, \ldots, z_n)$, $\mid z_h \mid \leq 1$, $h = 1, \ldots, n$, $i = 1, \ldots, n$.
We first relate $F_i(\cdot)$ to $G_{i-1}(\cdot)$, and subsequently $G_i(\cdot)$ to $F_i(\cdot)$. Thus we obtain an expression for $G_i(\cdot)$ in terms of $G_{i-1}(\cdot)$, which recursively yields a functional equation for $G_i(\cdot)$.
Define

$$d_i(z) := \sigma_i(\sum_{h=1}^{n} \lambda_h(1 - z_h)) \tag{9.18}$$

for $z = (z_1, \ldots, z_n)$, $\mid z_h \mid \leq 1$, $h = 1, \ldots, n$, $i = 1, \ldots, n$.

Then

$$F_i(z) = G_{i-1}(z)d_{i-1}(z), \tag{9.19}$$

where $d_0(\cdot)$, $G_0(\cdot)$ are to be understood as $d_n(\cdot)$, $G_n(\cdot)$ respectively.

Remark 9.4.1
In accordance with the model description, we assume here that the switch-over times are non-zero and that the servers keep switching when the system is empty. In case the switch-over times are zero, or in case the servers stop switching when the system is empty, (9.19) should be modified into

$$F_i(z) = G_{i-1}(z)d_{i-1}(z) + G_{i-1}(0)d_{i-1}(z)[e_i(z) - 1]$$

with

$$e_i(z) = \sum_{h \neq i} \frac{\lambda_h}{\lambda} z_h + \frac{\lambda_i}{\lambda}\theta_i(z).$$

Here $\theta_i(z)$ depends on what happens when an arriving type-i customer sees the servers idling at Q_i.

□

We now relate $G_i(\cdot)$ to $F_i(\cdot)$.

$$G_i(z) = \sum_{l_1=0}^{\infty} \cdots \sum_{l_n=0}^{\infty} \mathrm{E}(z_1^{\mathbf{Y}_{i1}} \cdots z_n^{\mathbf{Y}_{in}} \mid (\mathbf{X}_{i1}, \ldots, \mathbf{X}_{in}) = (l_1, \ldots, l_n)) \tag{9.20}$$
$$\Pr\{(\mathbf{X}_{i1}, \ldots, \mathbf{X}_{in}) = (l_1, \ldots, l_n)\}.$$

Evidently, it is the service discipline at Q_i that decides whether or not the right-hand side of (9.20) can be expressed into $F_i(\cdot)$. As discussed in Section 1.4, Fuhrmann [99] and Resing [159] consider the class of service disciplines (in single-server systems) that satisfy the following property:

Property 9.4.1
If there are k_i customers present at Q_i at the start of a visit, then during the course of the visit each of these k_i customers will effectively be replaced in an i.i.d. manner by a random population having pgf $h_i(z)$, which may be any n-dimensional pgf.
Formally,

$$\mathrm{E}(z_1^{\mathbf{Y}_{i1}} \cdots z_n^{\mathbf{Y}_{in}} \mid (\mathbf{X}_{i1}, \ldots, \mathbf{X}_{in}) = (l_1, \ldots, l_n)) = \tag{9.21}$$
$$z_1^{l_1} \cdots z_{i-1}^{l_{i-1}} (\eta_i(z))^{l_i} z_{i+1}^{l_{i+1}} \cdots z_n^{l_n}.$$

Substituting (9.21) into (9.20),

$$G_i(z) = F_i(z_1, \ldots, z_{i-1}, \eta_i(z), z_{i+1}, \ldots, z_n). \tag{9.22}$$

Using the theory of multi-type branching processes, both Fuhrmann and Resing show that the class of service disciplines that satisfy Property 9.4.1, like exhaustive and gated, allows a relatively simple exact analysis, basically due to the relatively simple form of (9.22). The results suggest that service disciplines that violate Property 9.4.1 defy an exact analysis, except for some special cases like two-queue cases and completely symmetric cases.

In multiple-server systems there are no non-trivial service disciplines that satisfy Property 9.4.1. However, some service disciplines *do* satisfy the following somewhat milder property than Property 9.4.1:

Property 9.4.2
If there are k_i customers present at Q_i at the start of a visit, then during the course of the visit one of these k_i customers will effectively be replaced by a random population having pgf $\eta_i^{(1)}(z)$, while each of the other customers will effectively be replaced in an i.i.d. manner by a random population having pgf $\eta_i(z)$.
Formally,

$$\mathrm{E}(z_1^{\mathbf{Y}_{i1}} \ldots z_n^{\mathbf{Y}_{in}} \mid (\mathbf{X}_{i1}, \ldots, \mathbf{X}_{in}) = (l_1, \ldots, l_n)) = \tag{9.23}$$
$$z_1^{l_1} \ldots z_{i-1}^{l_{i-1}} \eta_i^{(l_i)}(z) z_{i+1}^{l_{i+1}} \ldots z_n^{l_n},$$

with

$$\eta_i^{(l_i)}(z) = 1, \qquad l_i = 0, \tag{9.24}$$

$$\eta_i^{(l_i)}(z) = \eta_i^{(1)}(z)(\eta_i(z))^{l_i - 1}, \qquad l_i > 0. \tag{9.25}$$

Below we will describe some multiple-server systems with service disciplines that satisfy Property 9.4.2. We will then also briefly indicate some circumstances that may occur in single-server systems under the influence of which in principle simple service disciplines violate Property 9.4.1 but still satisfy Property 9.4.2.

Substituting (9.23), (9.24), (9.25) into (9.20),

$$G_i(z) = F_i(z_1, \ldots, z_{i-1}, \eta_i(z), z_{i+1}, \ldots, z_n)\frac{\eta_i^{(1)}(z)}{\eta_i(z)} + F_i(z_1, \ldots, z_{i-1}, 0, z_{i+1}, \ldots, z_n)[1 - \frac{\eta_i^{(1)}(z)}{\eta_i(z)}]. \tag{9.26}$$

Define

$$a_i(z) := (z_1, \ldots, z_{i-1}, \eta_i(z), z_{i+1}, \ldots, z_n); \tag{9.27}$$

$$b_i(z) := (z_1, \ldots, z_{i-1}, 0, z_{i+1}, \ldots, z_n); \tag{9.28}$$

$$c_i(z) := \frac{\eta_i^{(1)}(z)}{\eta_i(z)} \tag{9.29}$$

for $z = (z_1, \ldots, z_n)$, $\mid z_h \mid \leq 1$, $h = 1, \ldots, n$, $i = 1, \ldots, n$.
Then (9.26) may be written as

$$G_i(z) = F_i(a_i(z))c_i(z) + F_i(b_i(z))[1 - c_i(z)]. \tag{9.30}$$

In view of the results of Fuhrmann and Resing, one can in general not expect that the class of service disciplines that satisfy Property 9.4.2 but not Property 9.4.1, i.e., with $c_i(z) \neq 1$, allows an exact analysis, except possibly for some special cases. In the next section we will identify some of those cases.

We now describe some multiple-server systems with service disciplines that satisfy Property 9.4.2. Property 9.4.2 says that during the course of a visit to Q_i one of the customers initially present gets replaced by a different population than all the others. This suggests that either only one of the customers initially present at Q_i actually gets served or that all of them get served but that one of them keeps the servers busy for a different time than all the others. Keeping this in mind, we consider a class of service disciplines that are parametrized by two vectors $(p_1, \ldots, p_n)$ and $(q_1, \ldots, q_n)$ with the following interpretation. If there are any customers present at Q_i at the start of a visit, then one of them is served anyway, while the others are served with probability q_i. Customers arriving at Q_i during the course of a visit are served with probability p_i. The case $q_i = 0$ contains both the semi-exhaustive service discipline ($p_i = 1$) and the 1-limited service discipline ($p_i = 0$). The case $q_i = 1$ includes both the exhaustive service discipline ($p_i = 1$) and the gated service discipline ($p_i = 0$).
Denote $\kappa_i := \lambda_i p_i$.
Let $\mathbf{T}_i^{(k)}$ be the length of a busy period starting with k customers present in an ordinary $M/G/m$ queue with arrival rate κ_i and service time distribution $B_i(\cdot)$. Let $\tau_i^{(k)}(\omega) = \mathrm{E}(\mathrm{e}^{-\omega \mathbf{T}_i^{(k)}})$ for $\mathrm{Re}\,\omega \geq 0$.
Define

$$\alpha_i(z) := \sum_{h \neq i} \lambda_h(1 - z_h) + \lambda_i(1 - p_i)(1 - z_i)$$

for $z = (z_1, \ldots, z_n)$, $| z_h | \leq 1$, $h = 1, \ldots, n$, $i = 1, \ldots, n$.
For the above-defined class of service disciplines

$$\eta_i^{(l_i)}(z) = \sum_{k_i=1}^{l_i} \binom{l_i - 1}{k_i - 1} q_i^{k_i - 1}(1 - q_i)^{l_i - k_i} \tau_i^{(k_i)}(\alpha_i(z)) z_i^{l_i - k_i}, \tag{9.31}$$

with the interpretation of $\eta_i^{(l_i)}(z)$ as in (9.25). For $q_i = 0$, (9.31) satisfies (9.24) and (9.25) in Property 9.4.2 with $\eta_i(z) = z_i$, $\eta_i^{(1)}(z) = \tau_i^{(1)}(\alpha_i(z))$. If $\tau_i^{(k_i)}(\cdot)$ is of the form

$$\tau_i^{(k_i)}(\omega) = 1, \qquad k_i = 0, \tag{9.32}$$

$$\tau_i^{(k_i)}(\omega) = \tau_i^{(1)}(\omega)(\tau_i(\omega))^{k_i - 1}, \qquad k_i > 0, \tag{9.33}$$

for some LST $\tau_i(\cdot)$, then (9.31) satisfies (9.24) and (9.25) also for $q_i > 0$, with $\eta_i(z) = q_i \tau_i(\alpha_i(z)) + (1 - q_i) z_i$, $\eta_i^{(1)}(z) = \tau_i^{(1)}(\alpha_i(z))$.
We now give two examples where $\tau_i^{(k_i)}(\cdot)$ is of the form (9.32), (9.33).

Example 9.4.1
Assume that the service times at Q_i are exponentially distributed with parameter $\mu_i = 1/\beta_i$. Let $\mathbf{U}_i^{(k,l)}$ be the time needed in an $M/M/m$ queue with arrival rate κ_i and service rate μ_i to reduce the queue length from k to l, $k \geq l$. Let $\phi_i^{(k,l)}(\omega) = \mathrm{E}(\mathrm{e}^{-\omega \mathbf{U}_i^{(k,l)}})$ for $\mathrm{Re}\, \omega \geq 0$, $k \geq l$.
So $\tau_i^{(k_i)}(\omega) = \phi_i^{(k_i,0)}(\omega)$.
Let $\mathbf{U}_i$ be the length of a busy period in an $M/M/1$ queue with arrival rate κ_i and service rate $m\mu_i$. Let $\phi_i(\omega) = \mathrm{E}(\mathrm{e}^{-\omega \mathbf{U}_i})$ for $\mathrm{Re}\, \omega \geq 0$.
Because of the properties of the exponential distribution

$$\phi_i^{(k,0)}(\omega) = \prod_{l=1}^{k} \phi_i^{(l,l-1)}(\omega),$$

$$\phi_i^{(l,l-1)}(\omega) = \phi_i(\omega), \qquad l \geq m.$$

The transforms $\phi_i(\cdot)$, $\phi_i^{(1,0)}(\cdot), \ldots, \phi_i^{(m-1,m-2)}(\cdot)$ may be determined by solving a set of linear equations, cf. Medhi [149] pp. 138-140. In particular,

$$\phi_i(\omega) = \frac{\kappa_i + \omega + m\mu_i - \sqrt{(\kappa_i + \omega + m\mu_i)^2 - 4m\kappa_i\mu_i}}{2\kappa_i},$$

$$\phi_i^{(m-1,m-2)}(\omega) = \frac{(m-1)\phi_i(\omega)}{m - \phi_i(\omega)}.$$

For the gated service discipline, i.e., $p_i = 0$,

$$\phi_i(\omega) = \frac{m\mu_i}{m\mu_i + \omega},$$

$$\phi_i^{(l,l-1)}(\omega) = \frac{l\mu_i}{l\mu_i + \omega}, \qquad l \leq m.$$

Summarizing, for $m = 1$, $\tau_i^{(k_i)}(\cdot)$ is of the form (9.32), (9.33) with $\tau_i(\omega) = \tau_i^{(1)}(\omega) = \phi_i(\omega)$. Also for $m = 2$, $\tau_i^{(k_i)}(\cdot)$ is of the form (9.32), (9.33) with $\tau_i(\omega) = \phi_i(\omega)$, $\tau_i^{(1)}(\omega) = \phi_i^{(1,0)}(\omega)$, $\phi_i^{(1,0)}(\omega) = \dfrac{\phi_i(\omega)}{2 - \phi_i(\omega)}$. For $m > 2$, $\tau_i^{(k_i)}(\cdot)$ is no longer of the form (9.32), (9.33).

□

Example 9.4.2
Assume that the service times at Q_i are deterministic.
Let $\mathbf{V}_i$ be the length of a busy period in an ordinary $M/D/m$ queue with arrival rate κ_i and service time β_i. Let $\psi_i(\omega) = \mathrm{E}(\mathrm{e}^{-\omega \mathbf{V}_i})$ for $\mathrm{Re}\,\omega \geq 0$.
For $m = \infty$, $\tau_i^{(k_i)}(\cdot)$ is of the form (9.32), (9.33) with $\tau_i(\omega) = 1$, $\tau_i^{(1)}(\omega) = \psi_i(\omega)$, $\psi_i(\omega) = \dfrac{(\omega + \lambda_i)\mathrm{e}^{-\beta_i(\omega+\lambda_i)}}{\omega + \lambda_i \mathrm{e}^{-\beta_i(\omega+\lambda_i)}}$, cf. Stadje [170].

□

Remark 9.4.2
There are also some circumstances that may occur in single-server systems under the influence of which in principle simple service disciplines violate Property 9.4.1 but still satisfy Property 9.4.2. An obvious example arises when one of the customers served during a visit, e.g. the first one or the last one, requires an exceptional service time. Another example is provided by a set-up time or a shut-down time that is only incurred when at least one customer is served during a visit.

□

9.5 The joint queue length distribution II

In the previous section we obtained under assumption of Property 9.4.2 a set of $2n$ equations (9.19), (9.30) involving the $2n$ pgf's $F_i(z)$, $G_i(z)$, $i = 1, \ldots, n$. In this section we identify some cases in which these pgf's can actually be solved from these equations. Obviously it suffices to find either $F_i(z)$ or $G_i(z)$ for an arbitrary i, as the remaining $F_i(z)$, $G_i(z)$, $i = 1, \ldots, n$, can then easily be found from (9.19), (9.30).
Substituting (9.19) into (9.30),

$$G_i(z) = G_{i-1}(a_i(z))d_{i-1}(a_i(z))c_i(z) \tag{9.34}$$
$$+ \; G_{i-1}(b_i(z))d_{i-1}(b_i(z))[1 - c_i(z)], \tag{9.35}$$

where $d_0(\cdot)$, $G_0(\cdot)$ are to be understood as $d_n(\cdot)$, $G_n(\cdot)$, respectively.

Applying (9.34) n times we obtain a functional equation for $G_i(\cdot)$.
For $n = 1$ we find, using the definitions (9.18), (9.27), (9.28), (9.29),

$$G(z) = G(\eta(z))\sigma(\lambda(1-\eta(z)))\frac{\eta^{(1)}(z)}{\eta(z)} + G(0)\sigma(\lambda)[1 - \frac{\eta^{(1)}(z)}{\eta(z)}]. \qquad (9.36)$$

Here (as well as in the sequel) the redundant indices are omitted.
For $n = 2$ we find

$$G_i(z) =$$

$$G_i(a_{i-1}(a_i(z)))d_{i-1}(a_{i-1}(a_i(z)))c_{i-1}(a_i(z))d_i(a_i(z))c_i(z) +$$

$$G_i(b_{i-1}(a_i(z)))d_{i-1}(b_{i-1}(a_i(z)))[1 - c_{i-1}(a_i(z))]d_i(a_i(z))c_i(z) +$$

$$G_i(a_{i-1}(b_i(z)))d_{i-1}(a_{i-1}(b_i(z)))c_{i-1}(b_i(z))d_i(b_i(z))[1 - c_i(z)] +$$

$$G_i(b_{i-1}(b_i(z)))d_{i-1}(b_{i-1}(b_i(z)))[1 - c_{i-1}(b_i(z))]d_i(b_i(z))[1 - c_i(z)].$$

Using the definitions (9.18), (9.27), (9.28), (9.29),

$$G_1(z_1, z_2) = \qquad (9.37)$$

$$G_1(\eta_1(z), \eta_2(\eta_1(z), z_2))\sigma_2\{\gamma(\eta_1(z), \eta_2(\eta_1(z), z_2))\}\sigma_1\{\gamma(\eta_1(z), z_2)\} \times$$

$$\frac{\eta_2^{(1)}(\eta_1(z), z_2)}{\eta_2(\eta_1(z), z_2)}\frac{\eta_1^{(1)}(z)}{\eta_1(z)} +$$

$$G_1(\eta_1(z), 0)\sigma_2\{\gamma(\eta_1(z), 0)\}\sigma_1\{\gamma(\eta_1(z), z_2)\}[1 - \frac{\eta_2^{(1)}(\eta_1(z), z_2)}{\eta_2(\eta_1(z), z_2)}]\frac{\eta_1^{(1)}(z)}{\eta_1(z)} +$$

$$G_1(0, \eta_2(0, z_2))\sigma_2\{\gamma(0, \eta_2(0, z_2))\}\sigma_1\{\gamma(0, z_2)\}\frac{\eta_2^{(1)}(0, z_2)}{\eta_2(0, z_2)}[1 - \frac{\eta_1^{(1)}(z)}{\eta_1(z)}] +$$

$$G_1(0, 0)\sigma_2\{\gamma(0, 0)\}\sigma_1\{\gamma(0, z_2)\}[1 - \frac{\eta_2^{(1)}(0, z_2)}{\eta_2(0, z_2)}][1 - \frac{\eta_1^{(1)}(z)}{\eta_1(z)}],$$

(similarly with the indices interchanged) with $\gamma(z_1, z_2) = \lambda_1(1-z_1)+\lambda_2(1-z_2)$.
Remember that $\eta_i(z)$ is an n-dimensional pgf so that $\mid \eta_i(z) \mid \leq 1$ for $z = (z_1, \ldots, z_n)$, $\mid z_h \mid \leq 1$, $h = 1, \ldots, n$, $i = 1, \ldots, n$.
In general we obtain a functional equation for $G_i(\cdot)$ containing 2^n arguments in the right-hand side. So, in accordance with the results of Fuhrmann and Resing, in general the functional equation cannot be solved. In fact solving the functional equation only stands a chance in cases where 'enough' of the 2^n arguments in the right-hand side reduce either to z or to a constant. We will now indicate some of those cases.

Case I. $n = 1$ *queue,* $\eta(z) = z$.
This covers the case $q = 0$ described in the previous section, i.e., only one of the customers present at the start of a visit is served, while customers arriving during the course of a visit are served with probability p.
Rewriting (9.36),

$$G(z)[z - \sigma(\lambda(1-z))\eta^{(1)}(z)] = G(0)\sigma(\lambda)[z - \eta^{(1)}(z)]. \tag{9.38}$$

Letting $z \to 1$ in (9.38),

$$G(0) = \frac{1}{\sigma(\lambda)} \frac{1 - (\eta^{(1)})'(1) - \lambda s}{1 - (\eta^{(1)})'(1)},$$

with $(\eta^{(1)})'(1) = \left.\dfrac{d\eta^{(1)}(z)}{dz}\right|_{z=1}$. Apparently the stability condition is $\lambda s + (\eta^{(1)})'(1) < 1$. Note that $\lambda s + (\eta^{(1)})'(1)$ is the mean increase of the queue length between the start of two successive visits when the system is not empty, which should indeed be less than 1 to ensure stability.

Case II. $n = 1$ *queue,* $\eta(z) \neq z$.
This covers the case $q > 0$ described in the previous section, i.e., one of the customers present at the start of a visit is served anyway, the others are served with probability $q > 0$, while customers arriving during the course of a visit are served with probability p, moreover assuming that there are either two servers and exponential service times, cf. Example 9.4.1 or an infinite number of servers and deterministic service times, cf. Example 9.4.2.
Writing $e(z) = \eta(z)$, $f(z) = \sigma(\lambda(1 - \eta(z)))\eta^{(1)}(z)/\eta(z)$, $g(z) = \sigma(\lambda)[1 - \eta^{(1)}(z)/\eta(z)]$ in (9.36),

$$G(z) = G(e(z))f(z) + G(0)g(z). \tag{9.39}$$

Define

$$\begin{aligned} e^{(0)}(z) &:= z; \\ e^{(k)}(z) &:= e(e^{(k-1)}(z)); \qquad k = 1, 2, \ldots, \end{aligned}$$

for $|\, z \,| \leq 1$.
Iterating (9.39) K times,

$$\begin{aligned} G(z) &= G(e^{(K+1)}(z)) \prod_{k=0}^{K} f(e^{(k)}(z)) \\ &+ G(0) \sum_{k=0}^{K} g(e^{(k)}(z)) \prod_{l=0}^{k-1} f(e^{(l)}(z)). \end{aligned} \tag{9.40}$$

The next lemma establishes the convergence of (9.40) for $K \to \infty$ under the condition $\eta'(1) < 1$.

Lemma 9.5.1
If $\eta'(1) < 1$ then
i. $\lim_{K\to\infty} e^{(K+1)}(z) = 1$ for all z with $|z| \leq 1$;
ii. $\prod_{k=0}^{\infty} f(e^{(k)}(z))$ converges for all z with $|z| \leq 1$;
iii. $\sum_{k=0}^{\infty} g(e^{(k)}(z)) \prod_{l=0}^{k-1} f(e^{(l)}(z))$ converges for all z with $|z| \leq 1$.

Proof
See Appendix 9.B. □

Apparently the stability condition is $\eta'(1) < 1$. Note that $\eta'(1)$ is the mean number of customers by which each of the customers present at the start of a visit, except one, gets replaced in the course of the visit, which should indeed be less than 1 to ensure stability. In Example 9.4.1, $\eta'(1) = \dfrac{(1-p)\rho}{2-p\rho} < 1$ iff $\rho < 2$, irrespective of p. In Example 9.4.2, $\eta'(1) = 0$, also irrespective of p.
If $\eta'(1) < 1$ then, letting $K \to \infty$ in (9.40),

$$G(z) = \prod_{k=0}^{\infty} f(e^{(k)}(z)) + G(0) \sum_{k=0}^{\infty} g(e^{(k)}(z)) \prod_{l=0}^{k-1} f(e^{(l)}(z)). \tag{9.41}$$

Putting $z = 0$ in (9.41),

$$G(0) = \frac{\prod_{k=0}^{\infty} f(e^{(k)}(0))}{1 - \sum_{k=0}^{\infty} g(e^{(k)}(0)) \prod_{l=0}^{k-1} f(e^{(l)}(0))}.$$

Case III. $n = 2$ queues, $\eta_i(z) = z_i$, $i = 1, 2$.
This covers the case $q_i = 0$ described in the previous section, i.e., only one of the customers present at Q_i at the start of a visit is served, while customers arriving at Q_i during the course of a visit are served with probability p_i, $i = 1, 2$. Equation (9.37) reduces to

$$G_1(z_1, z_2)[z_1 z_2 - \sigma_1(\gamma(z))\sigma_2(\gamma(z))\eta_1^{(1)}(z)\eta_2^{(1)}(z)] =$$

$$G_1(z_1, 0)\sigma_1(\gamma(z))\sigma_2(\gamma(z_1, 0))\eta_1^{(1)}(z)[z_2 - \eta_2^{(1)}(z)] +$$

$$G_1(0, z_2)\sigma_1(\gamma(0, z_2))\sigma_2(\gamma(0, z_2))[z_1 - \eta_1^{(1)}(z)]\eta_2^{(1)}(0, z_2) +$$

$$G_1(0, 0)\sigma_1(\gamma(0, z_2))\sigma_2(\gamma(0))[z_1 - \eta_1^{(1)}(z)][z_2 - \eta_2^{(1)}(0, z_2)].$$

For $p_i = 0$, i.e., $\eta_i^{(1)}(z) = \beta_i(\gamma(z))$, $i = 1, 2$, the problem of solving the above functional equation may be formulated as a boundary value problem, cf. Boxma & Groenendijk [42].

Case IV. $n = 2$ queues, $\eta_i(z)$, $\eta_i^{(1)}(z)$ do not depend on z_i, $i = 1, 2$.
This occurs in Example 9.4.1 for $p_i = 1$, $i = 1, 2$, i.e., two servers, exponential service times, exhaustive service.
If $\eta_i(z)$, $\eta_i^{(1)}(z)$ do not depend on z_i, $i = 1, 2$, then the complete right-hand side of (9.37) does not depend on z_i. In other words, $G_i(z)$ does not depend on z_i, reflecting that Q_i is empty at the completion of a visit to Q_i when $p_i = 1$. So equation (9.37) may be replaced by

$$H_1(z_2) = H_1(e_1(z_2))f_1(z_2) + H_1(\eta_2(0))g_1(z_2) + H_1(0)h_1(z_2), \tag{9.42}$$

with

$$\begin{aligned}
e_1(z_2) &:= \eta_2(\eta_1(z), z_2);\\
f_1(z_2) &:= \sigma_2\{\gamma(\eta_1(z), \eta_2(\eta_1(z), z_2))\}\sigma_1\{\gamma(\eta_1(z), z_2)\}\frac{\eta_2^{(1)}(\eta_1(z), z_2)}{\eta_2(\eta_1(z), z_2)}\frac{\eta_1^{(1)}(z)}{\eta_1(z)};\\
g_1(z_2) &:= \sigma_2\{\gamma(0, \eta_2(0, z_2))\}\sigma_1\{\gamma(0, z_2)\}\frac{\eta_2^{(1)}(0, z_2)}{\eta_2(0, z_2)}[1 - \frac{\eta_1^{(1)}(z)}{\eta_1(z)}];\\
h_1(z_2) &:= \sigma_2\{\gamma(\eta_1(z), 0)\}\sigma_1\{\gamma(\eta_1(z), z_2)\}[1 - \frac{\eta_2^{(1)}(\eta_1(z), z_2)}{\eta_2(\eta_1(z), z_2)}]\frac{\eta_1^{(1)}(z)}{\eta_1(z)}\\
&\quad + \sigma_2\{\gamma(0, 0)\}\sigma_1\{\gamma(0, z_2)\}[1 - \frac{\eta_2^{(1)}(0, z_2)}{\eta_2(0, z_2)}][1 - \frac{\eta_1^{(1)}(z)}{\eta_1(z)}];\\
H_1(z_2) &:= G_1(z_1, z_2).
\end{aligned}$$

Define

$$\begin{aligned}
e_1^{(0)}(y) &:= y;\\
e_1^{(k)}(y) &:= e_1(e_1^{(k-1)}(y)); \qquad k = 1, 2, \ldots,
\end{aligned}$$

for $|y| \leq 1$.
Iterating (9.42) K times, writing $z_2 = y$,

$$\begin{aligned}
H_1(y) &= H_1(e_1^{(K+1)}(y))\prod_{k=0}^{K} f_1(e_1^{(k)}(y)) \qquad (9.43)\\
&\quad + H_1(\eta_2(0))\sum_{k=0}^{K} g_1(e_1^{(k)}(y))\prod_{l=0}^{k-1} f_1(e_1^{(l)}(y))\\
&\quad + H_1(0)\sum_{k=0}^{K} h_1(e_1^{(k)}(y))\prod_{l=0}^{k-1} f_1(e_1^{(l)}(y)).
\end{aligned}$$

The next lemma establishes the convergence of (9.43) for $K \to \infty$ under the condition $e_1'(1) < 1$.

Lemma 9.5.2
If $e_1'(1) < 1$ then
i. $\lim_{K\to\infty} e_1^{(K+1)}(y) = 1$ for all y with $| y | \leq 1$;
ii. $\prod_{k=0}^{\infty} f_1(e_1^{(k)}(y))$ converges for all y with $| y | \leq 1$;
iii. $\sum_{k=0}^{\infty} g_1(e_1^{(k)}(y)) \prod_{l=0}^{k-1} f_1(e_1^{(l)}(y))$ converges for all y with $| y | \leq 1$;
iv. $\sum_{k=0}^{\infty} h_1(e_1^{(k)}(y)) \prod_{l=0}^{k-1} f_1(e_1^{(l)}(y))$ converges for all y with $| y | \leq 1$.

Proof
Similar to the proof of Lemma 9.5.1.

□

Apparently the stability condition is $e_1'(1) < 1$. Note that $e_1'(1)$ is the mean number of type-1 customers by which each of the type-1 customers present at the start of a cycle, except one, gets replaced during the course of the cycle. In Example 9.4.1 when $p_i = 1$, $e_1'(1) = \eta_1'(1)\,\eta_2'(1) = \frac{\rho_1}{2-\rho_1}\frac{\rho_2}{2-\rho_2} = \frac{\rho_1\rho_2}{4-2\rho+\rho_1\rho_2} < 1$ iff $\rho < 2$.
If $e_1'(1) < 1$ then, letting $K \to \infty$ in (9.43),

$$\begin{aligned} H_1(y) &= \prod_{k=0}^{\infty} f_1(e_1^{(k)}(y)) \\ &+ H_1(\eta_2(0)) \sum_{k=0}^{\infty} g_1(e_1^{(k)}(y)) \prod_{l=0}^{k-1} f_1(e_1^{(l)}(y)) \\ &+ H_1(0) \sum_{k=0}^{\infty} h_1(e_1^{(k)}(y)) \prod_{l=0}^{k-1} f_1(e_1^{(l)}(y)). \end{aligned} \tag{9.44}$$

Putting $y = 0$ and $y = \eta_2(0)$ in (9.44) we obtain a pair of linear equations for the unknown constants $H_1(0)$ and $H_1(\eta_2(0))$.

Case V. $n = 2$ queues, $\eta_i(z) = 1$, $i = 1, 2$.
This occurs in Example 9.4.2, i.e., an infinite number of servers, deterministic service times, all the customers present at the start of a visit are served, while customers arriving during the course of a visit are served with probability p_i, $i = 1, 2$.
Equation (9.37) reduces to

$$G_1(z_1, z_2) = \tag{9.45}$$

$$\sigma_1(\lambda_2(1 - z_2))\eta_2^{(1)}(1, z_2)\eta_1^{(1)}(z) +$$

$$G_1(1,0)\sigma_2(\lambda_2)\sigma_1(\lambda_2(1-z_2))[1-\eta_2^{(1)}(1,z_2)]\eta_1^{(1)}(z)+$$

$$G_1(0,1)\sigma_2(\lambda_1)\sigma_1(\lambda_1+\lambda_2(1-z_2))\eta_2^{(1)}(0,z_2)[1-\eta_1^{(1)}(z)]+$$

$$G_1(0,0)\sigma_2(\lambda_1+\lambda_2)\sigma_1(\lambda_1+\lambda_2(1-z_2))[1-\eta_2^{(1)}(0,z_2)][1-\eta_1^{(1)}(z)].$$

Putting $z=(1,0)$, $z=(0,1)$, and $z=(0,0)$ in (9.45), we obtain a set of three linear equations for the unknown constants $G_1(1,0)$, $G_1(0,1)$, and $G_1(0,0)$.

Case VI. $n=2$ *queues,* $\eta_1(z)=z_1$, $\eta_2(z)=1$.
This covers the case $q_1=0$, $q_2=1$ described in the previous section, i.e., one of the customer present at Q_1 at the start of a visit is served, customers arriving at Q_1 during the course of a visit are served with probability p_1, all the customers present at Q_2 at the start of a visit are served, customers arriving at Q_2 during the course of a visit are served with probability p_2, moreover assuming that there are an infinite number of servers and deterministic service times at Q_2, cf. Example 9.4.2.
Equation (9.37) reduces to

$$G_1(z_1,z_2)= \tag{9.46}$$

$$G_1(z_1,1)\sigma_2(\gamma(z_1,1))\sigma_1(\gamma(z))\eta_2^{(1)}(z)\frac{\eta_1^{(1)}(z)}{z_1}+$$

$$G_1(z_1,0)\sigma_2(\gamma(z_1,0))\sigma_1(\gamma(z))[1-\eta_2^{(1)}(z)]\frac{\eta_1^{(1)}(z)}{z_1}+$$

$$G_1(0,1)\sigma_2(\gamma(0,1))\sigma_1(\gamma(0,z_2))\eta_2^{(1)}(0,z_2)[1-\frac{\eta_1^{(1)}(z)}{z_1}]+$$

$$G_1(0,0)\sigma_2(\gamma(0,0))\sigma_1(\gamma(0,z_2))[1-\eta_2^{(1)}(0,z_2)][1-\frac{\eta_1^{(1)}(z)}{z_1}].$$

Setting $z_2=0$ and $z_2=1$ in (9.46) we find expressions for $G_1(z_1,0)$ and $G_1(z_1,1)$ containing the unknown constants $G_1(0,0)$ and $G_1(0,1)$. Putting $z_1=0$ in those expressions we obtain a pair of linear equations for these constants.

Case VII. general n, $\eta_i(z)=1$, $i=1,\ldots,n$.
Similar to Case V.

Case VIII. general n, $\eta_1(z)=z_1$, $\eta_i(z)=1$, $i\neq 1$.
Similar to Case VI.

9.6 Concluding remarks and suggestions for further research

So far we focused on the joint queue length distribution at embedded epochs. In Section 9.4 we have obtained under assumption of Property 9.4.2 a set of $2n$ equations (9.19), (9.30) for the associated pgf's $F_i(z)$, $G_i(z)$, $i = 1, \ldots, n$. In Section 9.5 we have identified some cases in which these pgf's can actually be solved from these equations. These cases include several single-queue systems with a varying number of servers, two-queue two-server systems with exhaustive service and exponential service times, as well as infinite-server systems with an arbitrary number of queues, exhaustive or gated service, and deterministic service times.

To conclude, we now briefly discuss the derivation of the marginal queue length distribution at an arbitrary epoch from the joint queue length distribution at embedded epochs. Denote by $\mathbf{N}_i$ the queue length at Q_i at an arbitrary epoch. As stated in the introduction, in isolation a particular queue in a polling system may be viewed as a single-queue system with service interruptions, the intervisit periods constituting the service interruptions. In Section 9.2 we have shown how in such a system with service interruptions and exponential service times, the queue length distribution at an arbitrary epoch may be expressed into the queue length distribution at the beginning and the end of a service interruption. In case the assumptions of Section 9.2 are satisfied, one may thus obtain the marginal queue length distribution at Q_i from the queue length distribution at the beginning and the end of a visit to Q_i, given by $\mathrm{E}(z^{\mathbf{X}_{ii}}) = F_i(1, \ldots, 1, z, 1, \ldots, 1)$ and $\mathrm{E}(z^{\mathbf{Y}_{ii}}) = G_i(1, \ldots, 1, z, 1, \ldots, 1)$, respectively, with z as i-th argument. Consider e.g. the two-queue two-server system with exhaustive service and exponential service times, for which we obtained $F_i(z)$ and $G_i(z)$ in Case IV of the previous section. For such a system, using Lemma 9.2.1 and Lemma 9.2.2,

$$\mathrm{E}(z^{\mathbf{N}_i}) = \left[\frac{2}{2-\rho_i} + \frac{\rho_i}{2-\rho_i}\Pr\{\mathbf{N}_{i|I} = 0\}\right]^{-1} \times \left[\frac{2}{2-\rho_i z}\mathrm{E}(z^{\mathbf{N}_{i|I}}) + \frac{\rho_i z}{2-\rho_i z}\Pr\{\mathbf{N}_{i|I} = 0\}\right],$$

with

$$\mathrm{E}(z^{\mathbf{N}_{i|I}}) = \frac{1 - \mathrm{E}(z^{\mathbf{X}_{ii}})}{(1-z)\mathrm{E}\mathbf{X}_{ii}}.$$

In Section 9.2 we have also shown how subsequently the waiting-time distribution may be related to the marginal queue length distribution by using Lemma 9.2.3. In case the assumptions of Section 9.2 are not satisfied, one may quite often still obtain the marginal queue length distribution from the joint queue length distribution at the beginning and the end of a visit by developing ad hoc methods. We do however not pursue the matter any further, leaving it as an interesting topic for further research.

APPENDICES

9.A PROOF OF LEMMA 9.2.1

Lemma 9.2.1

$$\mathrm{E}(z^{\mathbf{N}}) =$$

$$\gamma \left[\sum_{l=0}^{m-2} \frac{\mathrm{E}(z^{\mathbf{N}^{(l)}_{M/M/m}}) \Pr\{\mathbf{N}_I = l\}}{\Pr\{\mathbf{N}^{(l)}_{M/M/m} = l\}} + \frac{m}{m - \rho z} \sum_{l=m-1}^{\infty} z^l \Pr\{\mathbf{N}_I = l\} \right],$$

with

$$\gamma = \left[\sum_{l=0}^{m-2} \frac{\Pr\{\mathbf{N}_I = l\}}{\Pr\{\mathbf{N}^{(l)}_{M/M/m} = l\}} + \frac{m}{m - \rho} \sum_{l=m-1}^{\infty} \Pr\{\mathbf{N}_I = l\} \right]^{-1}.$$

Proof

Define a vacation customer to be a customer arriving in a non-serving interval. Consider now a vacation customer C arriving at some time u. Suppose that C sees l customers upon arrival; so the queue length just after u equals $l+1$, $l \geq 0$. Let $\mathbf{T}$ be the first epoch in a serving interval after u at which the queue length reaches the level $l+1$ again. Let $\mathbf{U}$ be the first epoch after u at which the queue length drops to the level l. Suppose that the interval $[\mathbf{T}, \mathbf{U}]$ contains $\mathbf{K}$ distinct non-serving intervals starting at the consecutive epochs $\mathbf{U}_1, \ldots, \mathbf{U}_{\mathbf{K}}$, $\mathbf{K} \geq 0$. Let $\mathbf{N}_k$ be the queue length just after the epoch $\mathbf{U}_k$. Let $\mathbf{T}_k$ be the first epoch in a serving interval after $\mathbf{U}_k$ at which the queue length reaches the level $\mathbf{N}_k$ again. The interval $[\mathbf{T}, \mathbf{U}]$, exclusive of the intervals $[\mathbf{U}_1, \mathbf{T}_1], \ldots, [\mathbf{U}_{\mathbf{K}}, \mathbf{T}_{\mathbf{K}}]$, is called a fundamental period at level l. Note that we have thus established a 1-1 correspondence between fundamental periods at level l and vacation customers that see l customers upon arrival. (For $m = 1$ one can establish the 1-1 correspondence in an elegant way by choosing the order of service to be non-preemptive LCFS, cf. Fuhrmann & Cooper [102]; the vacation customer is then the 'ancestor' of the customers served in the fundamental period. For $m > 1$ one cannot establish the 1-1 correspondence in such an elegant way, as the customers then do not necessarily leave in order of service.) The notion of a fundamental period is illustrated in Figure 9.1, with $\mathbf{N}(t)$ denoting the queue length at time t. Parallel to the time axis the non-serving intervals are indicated by dotted lines. The serving intervals constituting a fundamental period at level $l = 1$ are indicated by bold lines.

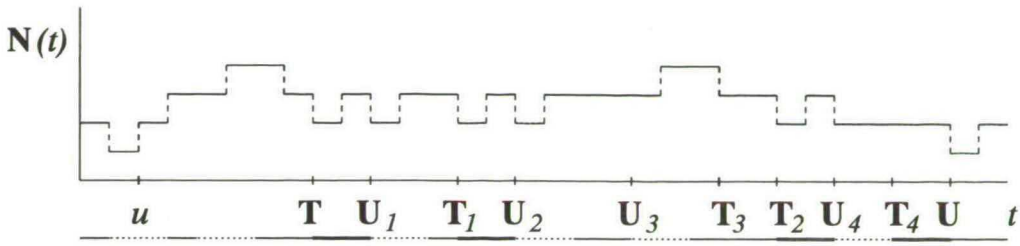

FIGURE 9.1. A fundamental period at level $l = 1$.

Consider now an arbitrary tagged customer as it departs from the system. Denote by $\mathbf{N}_D$ the number of customers that the tagged customer leaves behind. By virtue of the PASTA property and an up- & down crossing argument, $\mathbf{N}_D$ has the same distribution as $\mathbf{N}$. Denote by $\mathbf{L}_D$ the level of the fundamental period in which the tagged customer is served. (Note here that the fundamental periods together constitute a partitioning of the serving intervals.)

$$\mathrm{E}(z^{\mathbf{N}}) = \mathrm{E}(z^{\mathbf{N}_D}) = \sum_{l=0}^{\infty} \mathrm{E}(z^{\mathbf{N}_D} \mid \mathbf{L}_D = l)\mathrm{Pr}\{\mathbf{L}_D = l\}. \tag{9.47}$$

Define a fundamental period at level l in the corresponding $M/M/m$ queue to be a period ranging from an epoch when the queue length jumps to the level $l+1$ to an epoch when the queue length drops to the level l. Because of the memoryless property of the exponential service time distribution, a fundamental period at level l in the queue with service interruptions is stochastically indistinguishable from a fundamental period at level l in the corresponding $M/M/m$ queue. So, given that $\mathbf{L}_D = l$, $\mathbf{N}_D$ has the same distribution as the number of customers that an arbitrary customer leaves behind as it departs from the corresponding $M/M/m$ queue in a fundamental period at level l. By virtue of the (conditional) PASTA property and an up- & down crossing argument, this number has again the same distribution as $\mathbf{N}^{(l)}_{M/M/m}$, the queue length at an arbitrary epoch in the corresponding $M/M/m$ queue given that the queue length is at least l.

$$\mathrm{E}(z^{\mathbf{N}_D} \mid \mathbf{L}_D = l) = \mathrm{E}(z^{\mathbf{N}^{(l)}_{M/M/m}}). \tag{9.48}$$

Denote by $\mathbf{L}$ the level of an arbitrary fundamental period. Remember that we have established a 1-1 correspondence between fundamental periods at level l and vacation customers that see l customers upon arrival. So $\mathbf{L}$ has the same distribution as the number of customers seen by an arbitrary arriving vacation customer. Because of the PASTA property, this number has again the same distribution as $\mathbf{N}_I$. Denote by $\mathbf{M}_l$ the number of customers served in a fundamental period at level l. Then

$$\mathrm{Pr}\{\mathbf{L}_D = l\} = \frac{\mathrm{Pr}\{\mathbf{L} = l\}\mathrm{E}\mathbf{M}_l}{\sum_{k=0}^{\infty} \mathrm{Pr}\{\mathbf{L} = k\}\mathrm{E}\mathbf{M}_k} = \frac{\mathrm{Pr}\{\mathbf{N}_I = l\}\mathrm{E}\mathbf{M}_l}{\sum_{k=0}^{\infty} \mathrm{Pr}\{\mathbf{N}_I = k\}\mathrm{E}\mathbf{M}_k}. \tag{9.49}$$

In a fundamental period at level l exactly 1 customer is served that leaves behind l customers as it departs from the system. So EM_l equals the reciprocal of the probability that an arbitrary customer leaves behind l customers as it departs from the system in a fundamental period at level l:

$$\mathrm{EM}_l = \frac{1}{\Pr\{\mathbf{N}^{(l)}_{M/M/m} = l\}}. \tag{9.50}$$

Summarizing,

$$\mathrm{E}(z^{\mathbf{N}}) = \gamma \left[\sum_{l=0}^{\infty} \frac{\mathrm{E}(z^{\mathbf{N}^{(l)}_{M/M/m}}) \Pr\{\mathbf{N}_I = l\}}{\Pr\{\mathbf{N}^{(l)}_{M/M/m} = l\}} \right], \tag{9.51}$$

with

$$\gamma = \left[\sum_{k=0}^{\infty} \frac{\Pr\{\mathbf{N}_I = k\}}{\Pr\{\mathbf{N}^{(k)}_{M/M/m} = k\}} \right]^{-1}. \tag{9.52}$$

Substituting (9.2) into (9.51) and (9.52) completes the proof.

□

9.B Proof of Lemma 9.5.1

Lemma 9.5.1
If $\eta'(1) < 1$ then

i. $\lim_{K \to \infty} e^{(K+1)}(z) = 1$ for all z with $\mid z \mid \leq 1$;

ii. $\prod_{k=0}^{\infty} f(e^{(k)}(z))$ converges for all z with $\mid z \mid \leq 1$;

iii. $\sum_{k=0}^{\infty} g(e^{(k)}(z)) \prod_{l=0}^{k-1} f(e^{(l)}(z))$ converges for all z with $\mid z \mid \leq 1$.

Proof
Proof of i.
Since $e(z) = \eta(z)$ is a pgf,

$$|1 - e(z)| \leq \eta'(1) \mid 1 - z \mid . \tag{9.53}$$

By induction,

$$|1 - e^{(k)}(z)| \leq (\eta'(1))^k \, |1 - z|, \qquad k \geq 0. \tag{9.54}$$

So $\lim_{K \to \infty} e^{(K+1)}(z) = 1$.

Proof of ii.

According to the theory of infinite products, cf. Titchmarsh [185] p. 18, $\prod_{k=0}^{\infty} f(e^{(k)}(z))$ converges iff $\sum_{k=0}^{\infty}[1 - f(e^{(k)}(z))]$ converges.

Let $\Gamma(z)$ be the straight contour in the complex plane from z to 1.
According to the theory of complex functions,

$$|1 - f(z)| = |f(1) - f(z)| = |\int_{u\in\Gamma(z)} df(u)| \leq M(z)\,|1 - z|, \tag{9.55}$$

with

$$M(z) = \max_{u\in\Gamma(z)} |\frac{df(u)}{du}| < \infty,$$

as $f(u)$ is continuously-differentiable on $|u| \leq 1$.
Using (9.54), (9.55),

$$\sum_{k=0}^{\infty} |1 - f(e^{(k)}(z))| \leq \sum_{k=0}^{\infty} M(z)(\eta'(1))^k\,|1 - z| = \frac{M(z)}{1 - \eta'(1)}\,|1 - z| < \infty.$$

So $\prod_{k=0}^{\infty} f(e^{(k)}(z))$ converges.

Proof of iii.

Note that $\sum_{k=0}^{\infty} |g(e^{(k)}(z)) \prod_{l=0}^{k-1} f(e^{(l)}(z))| \leq L \sum_{k=0}^{\infty} |g(e^{(k)}(z)|$ with $L = \max_{k\geq 0} |\prod_{l=0}^{k-1} f(e^{(l)}(z))|$.

As $\prod_{k=0}^{\infty} f(e^{(k)}(z))$ converges, $\max_{k\geq 0} |\prod_{l=0}^{k-1} f(e^{(l)}(z))| < \infty$. So to prove that $\sum_{k=0}^{\infty} g(e^{(k)}(z))$ $\prod_{l=0}^{k-1} f(e^{(l)}(z))$ converges, it suffices to prove that $\sum_{k=0}^{\infty} g(e^{(k)}(z))$ converges.

Let $\Gamma(z)$ be the straight contour in the complex plane from z to 1.
Similarly to (9.55), noting that $g(1) = 0$,

$$|g(z)| \leq N(z)\,|1 - z|, \tag{9.56}$$

with

$$N(z) = \max_{u\in\Gamma(z)} |\frac{dg(u)}{du}| < \infty,$$

as $g(u)$ is also continuously-differentiable on $|u| \leq 1$.
Using (9.54), (9.56),

$$\sum_{k=0}^{\infty} |g(e^{(k)}(z))| \leq \sum_{k=0}^{\infty} N(z)(\eta'(1))^k\,|1 - z| = \frac{N(z)}{1 - \eta'(1)}\,|1 - z| < \infty.$$

So $\sum_{k=0}^{\infty} g(e^{(k)}(z)) \prod_{l=0}^{k-1} f(e^{(l)}(z))$ converges.

□

Chapter 10

Waiting-time approximations for multiple-server polling systems

10.1 Introduction

In the previous chapter we analyzed multiple-server polling systems in which the servers are assumed to be *coupled*, i.e., the servers visit the queues always together. In the present chapter we consider systems in which the servers are assumed to be *independent*, i.e., each of the servers visits the queues according to its own cyclic schedule. We derive waiting-time approximations for such systems with the exhaustive and gated service discipline.

As mentioned in the previous chapter, polling systems with multiple servers have received remarkably little attention in the vast literature on polling systems. One of the first studies is Morris & Wang [153]. They obtain the mean cycle time of each server and the mean intervisit time to a queue, and derive approximate expressions for the mean sojourn time for both a gated-type and a limited-type service discipline. A very interesting phenomenon observed by Morris & Wang is the tendency for the servers to cluster if they follow identical routes, especially in heavy traffic. Numerical experiments indicate that the bunching of servers is likely to deteriorate the system performance. Obviously the bunching of servers is alleviated if they follow different routes. Therefore Morris & Wang advocate the use of 'dispersive' schedules to improve the system performance.

In references [18], [121], [123], [157], [190] mean response time approximations are developed to analyze the performance of LAN's with multiple token rings. Mean response time approximations oriented to LAN's with a multiple slotted ring are contained in references [11], [18], [144], [190], [192]. Ajmone Marsan et al. [3], [4], [5] derive the mean cycle time and bounds for the mean waiting

times in symmetric systems for the exhaustive, gated, and 1-limited service discipline. In [2] they illustrate how Petri-net techniques may be used to study Markovian multiple-server polling systems.
Gamse & Newell [107] obtain approximate expressions for the mean round-trip time for a multiple-elevator facility. They make a comparison of some control options of multiple parallel elevators and - although there are some distinguishing features in the model description - find a similar tendency to form bunches as observed by Morris & Wang. For references on an exact analysis of models with a single queue or models with coupled servers we refer to the introduction of the previous chapter.
All these studies unanimously point out that multiple-server polling systems, combining the complexity of single-server polling systems and multiple-server systems, are extraordinarily hard to analyze. In fact, none of the studies (except [56], [57] for very specific two-queue infinite-server cases) presents any exact results for systems with multiple queues, apart from some mean-value results for global performance measures like cycle times.
In this chapter we consider the case of independent servers, each of which visits the queues according to its own cyclic schedule. In view of the mathematical intractability, we are interested in deriving waiting-time approximations for asymmetric systems with the exhaustive and gated service discipline. Most of the existing approximations are only developed for systems with the 1-limited service discipline or are restricted to completely symmetric systems. Moreover, most of the approximations completely ignore the considerable influence of the visit order on the waiting times or simply assume the visit order to be dispersive.
The remainder of the chapter is organized as follows. We present a detailed model description in Section 10.2. In Section 10.3 some preliminary results are obtained for the mean interarrival times of the various servers at the various queues, which will repeatedly be used throughout the next sections. In Sections 10.4, 10.5, 10.6 we derive waiting-time approximations for asymmetric systems with the exhaustive and gated service discipline. Considering the merits and drawbacks of existing approximations, we intend (i) to use pseudo-conservation-like concepts which have proven to be a very useful instrument in the single-server case, and (ii) to take into account the visit orders of the servers which in the multiple-server case, through the clustering effects, appear to have a major impact on the waiting times. In Section 10.7 the approximations are tested for a wide range of parameter combinations by comparison with either simulation results or exact numerical results obtained from the power-series algorithm (PSA), as elaborated upon in [150]. In Section 10.8 we conclude with some remarks and suggestions for further research.

10.2 MODEL DESCRIPTION

The model under consideration consists of n queues, $Q_1, \ldots, Q_n$, each of infinite capacity, attended by m identical servers, $S_1, \ldots, S_m$. For the specification of the arrival, service, and switch-over processes we refer to the description of the 'basic model' in Section 1.3.

The servers move from queue to queue in a cyclic manner. Server j visits the queues in the order $Q_{\pi_j(1)}, \ldots, Q_{\pi_j(n)}$, with $(\pi_j(1), \ldots, \pi_j(n))$ a permutation of $(1, \ldots, n)$, $j = 1, \ldots, m$.

The servers visit the queues independently of each other, under the restriction that at most m_i servers may visit Q_i simultaneously. In view of the latter restriction a server arrival will be called effective if there are less than m_i other servers already busy at Q_i. If an arrival at Q_i is not effective, then the server starts switching to the next queue immediately. If an arrival at Q_i *is* effective, then the server starts serving type-i customers (possibly none), as prescribed by the service discipline at Q_i. At each queue the service discipline may either be exhaustive or gated. Under the exhaustive service discipline a server leaves the queue when there are no waiting customers left. Under the gated service discipline a server leaves the queue when there are no waiting customers left whose arrival fell before the last server arrival. In other words, at each server arrival an imaginary gate opens to let waiting customers pass through. At each queue customers are taken into service in order of arrival. As soon as the server finishes serving type-i customers, as prescribed by the service discipline at Q_i, it starts switching to the next queue as specified in its schedule.

Finally some words on the stability conditions. Necessary conditions are of course that $\rho < m$, $\rho_i < m_i$, $i = 1, \ldots, n$. We strongly conjecture that these conditions are in fact also sufficient for service disciplines, like exhaustive and gated, that do not impose any (probabilistic) parametric restriction on the number of customers served during a server visit. Throughout the chapter the stability conditions are assumed to hold.

10.3 THE SERVER INTERARRIVAL TIME

In this section we derive some preliminary results for the mean interarrival time of the various servers at the various queues, which will repeatedly be used throughout the next sections in obtaining waiting-time approximations. We first introduce some notation. Define r_{ij} as the load carried by S_j at Q_i, i.e., the fraction of time that S_j is busy at Q_i, $i = 1, \ldots, n$, $j = 1, \ldots, m$. The total load carried by S_j is $r_j = \sum_{i=1}^{n} r_{ij}$. In general the fractions r_{ij} are unknown. However, the balance between the carried and the offered load at Q_i implies $\sum_{j=1}^{m} r_{ij} = \rho_i$, $i = 1, \ldots, n$.

Denote by $\mathbf{A}_{ij}$ ($\mathbf{A}_{ij}^*$) the interarrival (effective interarrival) time of S_j at Q_i, i.e., the time between two consecutive arrivals (effective arrivals) of S_j at Q_i,

$i = 1, \ldots, n$, $j = 1, \ldots, m$. Denote by p_{ij} the probability that an arbitrary arrival of S_j at Q_i is effective, i.e., the probability that at an arbitrary arrival of S_j there are less than m_i other servers already busy at Q_i, $i = 1, \ldots, n$, $j = 1, \ldots, m$. Obviously, for $m_i = m$, $p_{ij} = 1$; for $m_i < m$ the probabilities p_{ij} are however not known.
Applying a traffic balance argument,

$$\mathrm{E}\mathbf{A}_{ij} = \frac{s}{1 - r_j}, \tag{10.1}$$

independent of i.
As $p_{ij} = \mathrm{E}\mathbf{A}_{ij}/\mathrm{E}\mathbf{A}^*_{ij}$,

$$\mathrm{E}\mathbf{A}^*_{ij} = \frac{s/p_{ij}}{1 - r_j}. \tag{10.2}$$

A question that arises here quite naturally is whether or not all the servers will carry the same load. If two servers follow the same visit order, then by symmetry considerations both will also carry the same load of course. In particular, if all the servers follow the same visit order, then $r_{ij} = \rho_i/m$, $j = 1, \ldots, m$. Numerical experiments indicate that, even when the servers follow different visit orders, at each individual queue the load carried by each of the servers tends to differ only slightly, although in case of highly asymmetric system configurations the differences may increase somewhat. However, as observed in [150], [153], even in case of highly asymmetric system configurations the *total* load carried by each of the servers does not appear to differ significantly.
The above observation may be explained as follows. Suppose that the total load r_1 carried by S_1 is larger than the total load r_2 carried by S_2. So by (10.1) also the mean interarrival time $\mathrm{E}\mathbf{A}_{i1}$ of S_1 is larger than the mean interarrival time $\mathrm{E}\mathbf{A}_{i2}$ of S_2. In other words, S_2 visits the queues more frequently than S_1, so that S_2 is also likely (but not absolutely sure) to meet more work at the queues than S_1. So the total load r_2 carried by S_2 is likely to be larger than the total load r_1 carried by S_1, in contradiction with the initial supposition. The above explanation does not exclude that some minor differences may occur in the total load carried by each of the servers. However, the reasoning supports the observation that such differences cannot grow dramatically.
Denote by $\mathbf{A}_i$ ($\mathbf{A}^*_i$) the server interarrival (effective server interarrival) time at Q_i, i.e., the time between two consecutive server arrivals (effective server arrivals) at Q_i, $i = 1, \ldots, n$. Denote by p_i the probability that an arbitrary server arrival at Q_i is effective, i.e., the probability that at an arbitrary server arrival there are less than m_i other servers busy at Q_i, $i = 1, \ldots, n$. Obviously, for $m_i = m$, $p_i = 1$; for $m_i < m$ the probabilities p_i are however not known.
The $\mathbf{A}_i$ ($\mathbf{A}^*_i$) process is the superposition of the $\mathbf{A}_{ij}$ ($\mathbf{A}^*_{ij}$) processes. So,

$$\frac{1}{\mathrm{E}\mathbf{A}_i} = \sum_{j=1}^{m} \frac{1}{\mathrm{E}\mathbf{A}_{ij}} \qquad \left(\frac{1}{\mathrm{E}\mathbf{A}^*_i} = \sum_{j=1}^{m} \frac{1}{\mathrm{E}\mathbf{A}^*_{ij}} \right), \qquad i = 1, \ldots, n.$$

So, from (10.1),

$$\mathrm{E}\mathbf{A}_i = \frac{s}{m-\rho}, \tag{10.3}$$

which is again like (10.1) independent of i. Moreover, the mean server interarrival time is completely insensitive to how the total load is divided among the individual servers.
As $p_i = \mathrm{E}\mathbf{A}_i/\mathrm{E}\mathbf{A}_i^*$,

$$\mathrm{E}\mathbf{A}_i^* = \frac{s/p_i}{m-\rho}, \qquad i = 1, \ldots, n. \tag{10.4}$$

Note that the mean-value results obtained here for the (effective) server interarrival time also hold for the (effective) server interdeparture time. (A departure is called effective if it corresponds to an effective arrival.)

10.4 The waiting time

In this section we derive waiting-time approximations for systems with the exhaustive and gated service discipline. We first introduce some notation. Denote by $\mathbf{W}_i$ the waiting time of an arbitrary type-i customer, $i = 1, \ldots, n$. For any non-negative continuous stochastic variable $\mathbf{T}$, denote by $\mathbf{RT}$ a stochastic variable with as distribution the residual-lifetime distribution of $\mathbf{T}$, i.e.,

$$\Pr\{\mathbf{RT} < t\} = \frac{1}{\mathrm{E}\mathbf{T}} \int_{u=0}^{t} (1 - \Pr\{\mathbf{T} < u\}) du, \; t \geq 0.$$

For reference, we first briefly review the single-server case. The usual approach to obtain waiting-time approximations may be outlined as follows. To start with, one derives an (approximative) relationship of the form

$$\mathrm{E}\mathbf{W}_i \approx \gamma_i \mathrm{E}\mathbf{RC}_i, \qquad i = 1, \ldots, n, \tag{10.5}$$

with $\mathbf{C}_i$ either the interarrival or the interdeparture time at Q_i, depending on the service discipline at Q_i. The symbol γ_i represents some coefficient in terms of the system parameters, which reflects the influence of the service discipline at Q_i.
For the exhaustive service discipline,

$$\mathrm{E}\mathbf{W}_i = (1-\rho_i)\mathrm{E}\mathbf{RD}_i, \qquad i = 1, \ldots, n, \tag{10.6}$$

with $\mathbf{D}_i$ the server inter*departure* time at Q_i, cf. [88], [113]. (An alternative relationship for the exhaustive service discipline is $\mathrm{E}\mathbf{W}_i = \dfrac{\lambda_i \beta_i^{(2)}}{2(1-\rho_i)} + \mathrm{E}\mathbf{RI}_i$, with $\mathbf{I}_i$ the intervisit time at Q_i, cf. [62].)
For the gated service discipline,

$$\mathrm{E}\mathbf{W}_i = (1+\rho_i)\mathrm{E}\mathbf{RA}_i, \tag{10.7}$$

with, as before, $\mathbf{A}_i$ the server inter*arrival* time at Q_i, cf. [88], [113].
To proceed, one turns to approximating $\mathbf{ERC}_i$ (or $\mathbf{ERI}_i$). Since $\mathbf{ERC}_i = \mathrm{E}(\mathbf{C}_i)^2/2\mathbf{EC}_i$ (similarly for $\mathbf{ERI}_i$), cf. [95], where $\mathbf{EC}_i = s/(1-\rho)$ (respectively $\mathbf{EI}_i = (1-\rho_i)s/(1-\rho)$), it remains to approximate $\mathrm{E}(\mathbf{C}_i)^2$ (or $\mathrm{E}(\mathbf{I}_i)^2$) by using some additional information. One approach, followed by Bux & Truong [62] in the case of exhaustive service and deterministic switch-over times, is to derive an exact formula for $\mathrm{E}(\mathbf{I}_i^2)$ in the case of two queues, subsequently applying a 'heuristic extrapolation' to the case of an arbitrary number of queues. Another approach, proposed by Everitt [88] and further elaborated on by Groenendijk [113] is to approximate $\mathbf{ERC}_i$ in a direct manner, by invoking a so-called pseudo-conservation law, which provides an exact explicit expression for a weighted sum of the mean waiting times, typically $\sum_{i=1}^{n} \rho_i \mathbf{EW}_i$, cf. also Chapter 2. Substituting (10.5) into a pseudo-conservation law, assuming $\mathbf{ERC}_i \approx \mathbf{ERC}$, yields an approximation for $\mathbf{ERC}_i$. Note that the latter method in particular yields an exact expression for the mean waiting time in completely symmetric systems.

We now return to the multiple-server case. The usual approach to obtain waiting-time approximations in the multiple-server case may be sketched as follows, cf. [18], [121], [123], [153], [157], [190]. Like in the single-server case, one starts by deriving an (approximative) relationship of the form

$$\mathrm{E}\mathbf{W}_i \approx \gamma_i \mathrm{E}\mathbf{RC}_i^*, \tag{10.8}$$

with $\mathbf{C}_i^*$ either the *effective* server interarrival or interdeparture time at Q_i. At that stage the complications already start, as for most service disciplines at best a very rough approximation for γ_i can be found. Next, like in the single-server case, one proceeds by approximating $\mathbf{ERC}_i^*$. The complications then grow even worse, as there is very little additional information available that can be used, neither in the form of exact results for special cases, nor in the global form of a pseudo-conservation law. Having little choice left, one typically considers the $\mathbf{C}_i^*$-process as resulting from the $\mathbf{C}_i$-process after a 'filtering' with probability p_i (the probability of an arrival at Q_i being effective), and then the $\mathbf{C}_i$-process in its turn as the superposition of the $\mathbf{C}_{ij}$-processes, with j indicating server j. Subsequently one approximates p_i and fits some distribution to the $\mathbf{C}_{ij}$-processes, assuming that the $\mathbf{C}_{ij}$-processes are independent and identically distributed. The motivation for fitting some particular distribution to the $\mathbf{C}_{ij}$-processes is at best questionable, but is usually even completely lacking. What is worse however, is that the assumption that the $\mathbf{C}_{ij}$-processes are independent and identically distributed completely ignores the tendency for the servers to cluster, which immediately explains why the resulting approximations only appear to be reasonably accurate for dispersive schedules or under conditions (like $m_i = 1$, or 1-limited service) with dispersive effects, cf. [18], [121], [123], [153], [157], [190].

We now describe an alternative approach to derive waiting-time approximations. From now on we focus on the case $m_i = m$, which we consider to be the most interesting case; in the last section of the chapter we briefly discuss the case $m_i = 1$. Considering the above-mentioned objections, we intend

(i) to take into account the visit orders of the servers which in the multiple-server case, through the clustering effects, appear to have a major impact on the waiting times;
(ii) to avoid considering cycle-time processes, instead using pseudo-conservation-like concepts which have proven to be a very useful instrument in the single-server case.

Denote by q_i the steady-state probability that at least one of the servers is busy at Q_i. In general the probabilities q_i are unknown. However, $\rho_i/m \leq q_i \leq \min\{\rho_i, 1\}$. To derive an approximative relationship of the form $\mathbf{EW}_i \approx \gamma_i \mathbf{ERC}_i$, we assume that the customers experience the presence of multiple servers as if there is a single server processing at speed $\alpha_i = \rho_i/q_i$, the exact average processing speed at Q_i.
For the exhaustive service discipline we then obtain from (10.6), replacing ρ_i by ρ_i/α_i,

$$\mathbf{EW}_i \approx (1 - q_i)\mathbf{ERD}_i, \tag{10.9}$$

with $\mathbf{D}_i$ the server interdeparture time at Q_i.
Similarly, we obtain from (10.7) for the gated service discipline,

$$\mathbf{EW}_i \approx (1 + q_i)\mathbf{ERA}_i, \tag{10.10}$$

with $\mathbf{A}_i$, as before, the server interarrival time at Q_i.
In the multiple-server case it is no longer reasonable to assume that the residual server interdeparture (interarrival) times are approximately equal, as the degree of clustering may differ significantly from queue to queue. Instead we assume that the residual server interdeparture (interarrival) times are proportional to the average processing speed $\alpha_i = \rho_i/q_i$, which may be seen as a measure for the degree of clustering at Q_i, i.e.,

$$\mathbf{ERD}_i \approx \mathbf{ERD}\rho_i/q_i, \tag{10.11}$$

and

$$\mathbf{ERA}_i \approx \mathbf{ERA}\rho_i/q_i, \tag{10.12}$$

with **ERD** and **ERA** some unknown constants. Note that in case $m = 1$, $q_i = \rho_i$, so that (10.11) and (10.12) reduce to $\mathbf{ERD}_i \approx \mathbf{ERD}$ and $\mathbf{ERA}_i \approx \mathbf{ERA}$, respectively, the usual assumptions in the single-server case.
To complete the derivation of the approximations, it suffices to (i) find an expression for the weighted sum $\sum_{i=1}^{n} \rho_i \mathbf{EW}_i$ and (ii) determine the probabilities q_i, which we will do in Sections 10.5 and 10.6, respectively.

10.5 APPROXIMATING THE WEIGHTED SUM $\sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i$

In this section we describe a method for approximating $\sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i$. Denote by $\mathbf{V}$ the steady-state total amount of work in the system. Applying Brumelle's formula [60],

$$\sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i = \mathrm{E}\mathbf{V} - \frac{1}{2} \sum_{i=1}^{n} \lambda_i \beta_i^{(2)}. \tag{10.13}$$

So to find an expression for $\sum_{i=1}^{n} \rho_i \mathrm{E}\mathbf{W}_i$ it suffices to find an expression for the mean amount of work $\mathrm{E}\mathbf{V}$. For reference, we first briefly review the single-server case, where the crucial property that facilitates the determination of $\mathrm{E}\mathbf{V}$ is work decomposition, which in its turn builds on the fundamental property of work conservation, cf. also Chapter 2. To illuminate these concepts, denote by $\mathbf{V}^0$ the steady-state total amount of work in the 'corresponding $M/G/1$ system'. The 'corresponding $M/G/1$ system' is a single-server system with similar traffic characteristics but with zero switch-over times, i.e., without any interruptions by the switch-over process. Denote by $\mathbf{Y}$ the steady-state amount of work in the original system in a switching interval. Then the following *work decomposition* property holds:

$$\mathbf{V} \stackrel{d}{=} \mathbf{V}^0 + \mathbf{Y}, \tag{10.14}$$

with $\stackrel{d}{=}$ indicating equality in distribution. When the amount of work in a switching interval is always zero, we may recognize in (10.14) the underlying property of *work conservation*, which in fact holds even in sample-path sense. Note that $\mathrm{E}\mathbf{V}^0$ is simply known from the Pollaczek-Khintchine formula. For a broad class of service disciplines, including gated and exhaustive, $\mathrm{E}\mathbf{Y}$ may be determined along the lines of [38], [42]. Taking expectations in (10.14), substituting into (10.13), then yields a so-called *pseudo-conservation law* for the mean waiting times.

We now return to the multiple-server case, where deriving a pseudo-conservation law in an exact way involves serious complications. A simple interchange argument shows that in the multiple-server case a strict work conservation property in sample-path sense only holds if all the customers have the same (deterministic) service time. A weaker work conservation property in stochastic sense only holds if all the customers have the same service time distribution and the service discipline is regardless of the actual service times. Hence, since work conservation may be seen as the basis for work decomposition, it is not very likely that a property like (10.14) holds in the multiple-server case. Even if it were, we would face the problem that $\mathrm{E}\mathbf{V}^0$ is not known in the multiple-server case, not to mention the problem of determining $\mathrm{E}\mathbf{Y}$, so that the chances of deriving a pseudo-conservation law in an exact way appear to be negligible. In-

stead we therefore derive an approximative pseudo-conservation law. Although a work decomposition property probably does not hold, we can always write

$$\mathrm{E}\mathbf{V} = \mathrm{E}\mathbf{V}^0 + \mathrm{E}\mathbf{Y},$$

with $\mathbf{V}^0$ denoting the steady-state total amount of work in the 'corresponding $M/G/m$ system' and $\mathbf{Y}$ representing a stochastic variable whose mean satisfies the above equality, but which further remains unspecified. (The 'corresponding $M/G/m$ system' is defined analogously as in the single-server case.) To approximate $\mathrm{E}\mathbf{V}$, we consider two auxiliary single-server systems with similar characteristics, for which the work decomposition property *does* hold, viz:
(i). the 'λ/m system', i.e., a single-server system with identical characteristics, but with the arrival rate decreased by a factor m;
(ii). the 'β/m system', i.e., a single-server system with identical characteristics, but with the service rate increased by a factor m.
For these auxiliary systems we adopt the notational convention introduced for the original system.
Applying (10.14) to the two auxiliary systems,

$$\mathbf{V}_{\lambda/m} \stackrel{d}{=} \mathbf{V}^0_{\lambda/m} + \mathbf{Y}_{\lambda/m}, \quad \mathbf{V}_{\beta/m} \stackrel{d}{=} \mathbf{V}^0_{\beta/m} + \mathbf{Y}_{\beta/m}.$$

From the Pollaczek-Khintchine formula,

$$\mathrm{E}\mathbf{V}^0_{\lambda/m} = \mathrm{E}\mathbf{V}^0_{\beta/m} = \frac{1}{m} \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}}{2(1-\hat{\rho})},$$

with $\hat{\rho} = \rho/m$.
From [38], [42],

$$\mathrm{E}\mathbf{Y}_{\lambda/m} = \frac{1}{m}\mathrm{E}\mathbf{Y}_{\beta/m} = \hat{\rho}\frac{s^{(2)}}{2s} + \frac{s}{2(1-\hat{\rho})}\left[\hat{\rho}^2 - \sum_{i\in E}\hat{\rho}_i^2 + \sum_{i\in G}\hat{\rho}_i^2\right],$$

with $\hat{\rho}_i = \rho_i/m$. The symbols E and G indicate the index sets of the queues with the exhaustive and gated service discipline, respectively.
To approximate $\mathrm{E}\mathbf{V}^0$, we assume that the ratio of the mean amount of work in a multiple-server system and a single-server system with similar characteristics and proportional load, is rather insensitive to the service time distribution, i.e.,

$$\frac{\mathrm{E}\mathbf{V}^0}{\mathrm{E}\mathbf{V}^0_{\lambda/m}} = \frac{\mathrm{E}\mathbf{V}^0}{\mathrm{E}\mathbf{V}^0_{\beta/m}} \approx \gamma(\rho),$$

with $\gamma(\rho)$ denoting the known value of the ratio in question in case of identically exponentially distributed service times. In other words,

$$\mathrm{E}\mathbf{V}_0 \approx \gamma(\rho)\mathrm{E}\mathbf{V}^0_{\lambda/m} = \gamma(\rho)\mathrm{E}\mathbf{V}^0_{\beta/m},$$

with

$$\gamma(\rho) = \frac{(m-\rho)\sum_{l=0}^{m-1} l\frac{\rho^l}{l!} + \frac{\rho^m}{m!}m^2 + \frac{\rho^m}{m!}\frac{m\rho}{m-\rho}}{\rho\sum_{l=0}^{m-1}\frac{\rho^l}{l!} + \frac{\rho^m}{m!}\frac{m\rho}{m-\rho}}. \tag{10.15}$$

To approximate $\mathrm{E}\mathbf{Y}$, we assume

$$\mathrm{E}\mathbf{Y} \approx (1-\alpha)\zeta_{\lambda/m}(\rho)\mathrm{E}\mathbf{Y}_{\lambda/m} + \alpha\zeta_{\beta/m}(\rho)\mathrm{E}\mathbf{Y}_{\beta/m},$$

with α indicating whether the comparison with the 'λ/m system' or with the 'β/m system' is most appropriate. The interpretation of the coefficients $\zeta_{\lambda/m}(\rho)$ and $\zeta_{\beta/m}(\rho)$ is similar to that of the factor $\gamma(\rho)$ introduced above.
If the server clustering is strong, which will occur especially in heavy traffic if the servers follow identical routes, then the system will tend to behave as the 'β/m system', i.e., $\alpha \uparrow 1$ for a high degree of clustering. It also suggests choosing $\zeta_{\beta/m}(\rho) = 1$ (note from (10.15) that also $\gamma(\rho) \downarrow 1$ when $\rho \uparrow m$). On the other hand, if the server clustering is weak, which will occur in light traffic, or if a dispersive schedule is used, then the comparison with the 'λ/m system' is probably more appropriate, i.e., $\alpha \downarrow 0$ for a low degree of clustering. Choosing $\zeta_{\lambda/m}(\rho)$ in this case is however not so easy. In light traffic the switch-over times will tend to dominate the behavior of the system. In fact, if the total switch-over time incurred during a cycle is deterministic, $\mathrm{E}\mathbf{W}_i \downarrow s/(m+1)$ for $\rho \downarrow 0$. If the total switch-over time during a cycle is exponentially distributed, $\mathrm{E}\mathbf{W}_i \downarrow s/m$ for $\rho \downarrow 0$. Interpolating we obtain $\mathrm{E}\mathbf{W}_i \downarrow (m + s^{(2)}/s^2 - 1)s/m(m+1)$ for $\rho \downarrow 0$, implying that $\mathrm{E}\mathbf{Y} = \rho(m + s^{(2)}/s^2 - 1)s/m(m+1) + O(\rho^2)$ for $\rho \downarrow 0$. Note that $\mathrm{E}\mathbf{Y}_{\lambda/m} = \rho s^{(2)}/2ms + O(\rho^2)$ for $\rho \downarrow 0$. So

$$\mathrm{E}\mathbf{Y}/\mathrm{E}\mathbf{Y}_{\lambda/m} \to \frac{(m + s^{(2)}/s^2 - 1)s/m(m+1)}{s^{(2)}/2ms} = 2(1 + (m-1)s^2/s^{(2)})/(m+1)$$

for $\rho \downarrow 0$. In other words, $\zeta_{\lambda/m}(\rho) \to 2(1+(m-1)s^2/s^{(2)})/(m+1)$ for $\rho \downarrow 0$. On the other hand, in heavy traffic the switch-over times occupy only a negligible fraction of time, implying that $\zeta_{\lambda/m}(\rho) \to \gamma(\rho)$ for $\rho \uparrow m$. Interpolating we obtain $\zeta_{\lambda/m}(\rho) \approx 2(\rho/m)(1 + (m-1)s^2/s^{(2)})/(m+1) + \gamma(\rho)(1-\rho/m)$.
To choose α, we consider again $\alpha_i = \rho_i/q_i$, the average processing speed at Q_i, as a measure for the degree of clustering at Q_i. We define

$$\alpha := \frac{1}{n}\sum_{i=1}^{n} \frac{\alpha_i - \rho_i/(1-(1-\rho_i/m)^m)}{m - \rho_i/(1-(1-\rho_i/m)^m)}. \tag{10.16}$$

Note that for a high degree of clustering, i.e., $q_i \downarrow \rho_i/m$, $\alpha \uparrow 1$. On the other hand, for a low degree of clustering, i.e., $q_i \uparrow 1-(1-\rho_i/m)^m$, $\alpha \downarrow 0$.
Concluding,

$$\mathrm{E}\mathbf{V} \approx \frac{\gamma(\rho)}{m}\frac{\sum_{i=1}^{n}\lambda_i\beta_i^{(2)}}{2(1-\hat{\rho})} \tag{10.17}$$

$$+ \left((1-\alpha)\left(2\frac{\rho}{m}\frac{1+(m-1)s^2/s^{(2)}}{m+1}+\gamma(\rho)\left(1-\frac{\rho}{m}\right)\right)+\alpha m\right)$$
$$\times \left(\hat{\rho}\frac{s^{(2)}}{2s}+\frac{s}{2(1-\hat{\rho})}\left[\hat{\rho}^2-\sum_{i\in E}\hat{\rho}_i^2+\sum_{i\in G}\hat{\rho}_i^2\right]\right),$$

with $\gamma(\rho)$ and α as in (10.15) and (10.16), respectively. Substituting (10.17) into (10.13) yields an approximative pseudo-conservation law. Subsequently substituting (10.9), (10.10), (10.11), (10.12) into (10.13) yields waiting-time approximations, still containing the probabilities q_i, which we will determine in the next section. Note that the method in particular yields an exact expression for the mean waiting time in completely symmetric systems with exponential service times and zero switch-over times.

10.6 Approximating the probabilities q_i

In this section we describe a method for approximating the probabilities q_i that at least one of the servers is busy at Q_i, $i = 1, \ldots, n$. We first introduce some notation. Denote by $\mathbf{H}_j(t)$ the entry in the polling table of S_j at time t. Indicate by $\mathbf{Z}_j(t)$ whether S_j is switching ($\mathbf{Z}_j(t) = 0$) or serving ($\mathbf{Z}_j(t) = 1$) at time t. So if $(\mathbf{H}_j(t), \mathbf{Z}_j(t)) = (h, 0)$, then S_j is switching to $Q_{\pi_j(h)}$ at time t; if $(\mathbf{H}_j(t), \mathbf{Z}_j(t)) = (h, 1)$, then S_j is serving at $Q_{\pi_j(h)}$ at time t. Denote by $(\mathbf{H}, \mathbf{Z})$ a pair of stochastic variables with as joint distribution the joint stationary distribution of $(\mathbf{H}(t), \mathbf{Z}(t))$ with $(\mathbf{H}(t), \mathbf{Z}(t)) = (\mathbf{H}_1(t), \ldots, \mathbf{H}_m(t), \mathbf{Z}_1(t), \ldots, \mathbf{Z}_m(t))$.
We now describe a method for approximating the distribution of $(\mathbf{H}, \mathbf{Z})$. Note that the probabilities q_i follow immediately from the distribution of $(\mathbf{H}, \mathbf{Z})$ as

$$q_i = 1 - \Pr\{(\pi_j(\mathbf{H}_j), \mathbf{Z}_j) \neq (i, 1), j = 1, \ldots, m\}. \tag{10.18}$$

In fact it is not difficult to approximate each of the *marginal* distributions of $(\mathbf{H}_j, \mathbf{Z}_j)$, $j = 1, \ldots, m$. As observed in Section 10.3, at each individual queue the load carried by each of the servers tends to differ only rather slightly, i.e., $r_{ij} \approx \rho_i/m$, $i = 1, \ldots, n$, $j = 1, \ldots, m$. So

$$\Pr\{(\mathbf{H}_j, \mathbf{Z}_j) = (h, 1)\} = r_{\pi_j(h)j} \approx \frac{\rho_{\pi_j(h)}}{m}. \tag{10.19}$$

Also, from (10.1),

$$\Pr\{(\mathbf{H}_j, \mathbf{Z}_j) = (h, 0)\} = \frac{s_{\pi_j(h)}}{\mathrm{EC}_j} \approx (1 - \frac{\rho}{m})\frac{s_{\pi_j(h)}}{s}, \tag{10.20}$$

with EC_j denoting the mean cycle time of S_j.
However, it is considerably harder to approximate the *simultaneous* distribution of $(\mathbf{H}, \mathbf{Z}) = (\mathbf{H}_1, \ldots, \mathbf{H}_m, \mathbf{Z}_1, \ldots, \mathbf{Z}_m)$, which is actually needed in (10.18). There are three types of transitions in $(\mathbf{H}, \mathbf{Z})$.

First,

$$(h,z) \to (h+e_j, z-e_j), \qquad z_j = 1, \tag{10.21}$$

representing a departure of S_j from $Q_{\pi_j(h_j)}$ which does not result from an instantaneous passage; here e_j represents the j-th m-dimensional unit vector; $h_j + 1$ is in fact to be understood as $(h_j \bmod n) + 1$.
Second,

$$(h,z) \to (h, z+e_j), \qquad z_j = 0, \tag{10.22}$$

representing an arrival of S_j at $Q_{\pi_j(h_j)}$ which does not lead to an instantaneous passage.
Third,

$$(h,z) \to (h+e_j, z), \qquad z_j = 0, \tag{10.23}$$

representing an instantaneous passage of S_j at $Q_{\pi_j(h_j)}$.
Note that $\{(\mathbf{H}(t), \mathbf{Z}(t)), t \geq 0\}$ is *not* a Markov process, as the transitions are not independent of the past. To approximate the simultaneous distribution of $(\mathbf{H}, \mathbf{Z})$, we will however deal with the process as if it *is* Markov, i.e., as if the transitions in $(\mathbf{H}, \mathbf{Z})$ occur at a constant rate, independent of the past. The distribution of $(\mathbf{H}, \mathbf{Z})$ may then be determined, as soon as the transition rates $\mu_{(h,z)\to(h',z')}$ are specified, which we might do as follows.
First,

$$\mu_{(h,z)\to(h+e_j,z-e_j)} = (m-\rho)/(\rho_{\pi_j(h_j)} s), \qquad z_j = 1, \tag{10.24}$$

i.e., a departure of S_j from $Q_{\pi_j(h_j)}$ (which does not result from an instantaneous passage) occurs at a rate reciprocal to the approximate mean visit time of S_j at $Q_{\pi_j(h_j)}$ (i.e. $r_{\pi_j(h_j),j} \mathrm{E}\mathbf{A}_{\pi_j(h_j),j} \approx \rho_{\pi_j(h_j)} s/(m-\rho)$).
Second,

$$\mu_{(h,z)\to(h,z+e_j)} = 1/s_{\pi_j(h_j)}, \qquad z_j = 0, \tag{10.25}$$

i.e., an arrival of S_j at $Q_{\pi_j(h_j)}$ (which does not lead to an instantaneous passage) occurs at a rate reciprocal to the mean switch-over time into $Q_{\pi_j(h_j)}$.
Third,

$$\mu_{(h,z)\to(h+e_j,z)} = 0, \qquad z_j = 0, \tag{10.26}$$

i.e., an instantaneous passage of S_j at $Q_{\pi_j(h_j)}$ (only occurring when there are no waiting customers at $Q_{\pi_j(h_j)}$, which cannot be deduced from $(\mathbf{H}, \mathbf{Z})$) does not occur. Note that in light traffic instantaneous passages in fact *do* frequently occur, as a server arrival is likely be attended by a concurrent server departure, which might suggest to replace (10.26) by

$$\mu_{(h,z)\to(h+e_j,z)} = 1/s_{\pi_j(h_j)}, \qquad z_j = 0, \tag{10.27}$$

when $\rho \downarrow 0$. However, when $\rho \downarrow 0$ in light traffic, combining (10.25) with (10.24) has a similar effect as using (10.27) would have.
Because of the homogeneity in the transition rates we would obtain from (10.24), (10.25), (10.26)

$$\Pr\{(\mathbf{H}, \mathbf{Z}) = (h, z)\} = \prod_{j=1}^{m} \Pr\{(\mathbf{H}_j, \mathbf{Z}_j) = (h_j, z_j)\}, \tag{10.28}$$

where $\Pr\{(\mathbf{H}_j, \mathbf{Z}_j) = (h_j, z_j)\}$ satisfies (10.19), (10.20). In other words, we would obtain complete independence in the server position distribution, while in fact we attempted to capture the tendency for the servers to cluster.
The driving force behind the tendency for the servers to cluster is that the time that a server visits a queue depends on the time that the queue has not been visited by one of the other servers, so that the servers, somewhat depending on the visit orders, tend to be driven together. Once driven together, the servers do not disperse, as long as the visit orders do not direct them to different queues. To capture these phenomena, we slightly modify the transitions for the states in which more than one server is busy at the same queue simultaneously. For these states we replace the transitions where *one* server leaves the queue by a single transition of the same rate where *all* the visiting servers leave the queue simultaneously, reflecting that actually all the servers will tend to leave relatively shortly after one another.
The transition rates being specified, the distribution of $(\mathbf{H}, \mathbf{Z})$ may then be determined by solving the balance equations, supplemented with the normalization condition. Because of the inhomogeneity introduced in the transition rates it is no longer possible to give the simultaneous distribution as explicitly as in (10.28), but it is easily verified from the balance equations that the marginal distribution $\Pr\{(\mathbf{H}_j, \mathbf{Z}_j) = (h_j, z_j)\}$ still satisfies (10.19), (10.20).

More detailed clustering measures
Remember that we approximated the distribution of $(\mathbf{H}, \mathbf{Z})$ in the first place to determine the probabilities q_i that at least one of the servers is busy at Q_i, $i = 1, \ldots, n$. In their turn we used the probabilities q_i to determine $\alpha_i = \rho_i / q_i$, the average processing speed at Q_i, as a measure for the degree of clustering at Q_i. Having approximated the simultaneous distribution of $(\mathbf{H}, \mathbf{Z})$, we may however refine the latter estimate for the degree of clustering, to deal with situations in which the average processing speed does not provide a good indication for the degree of clustering (e.g. in case of a lightly-loaded queue preceded by a heavily-loaded queue). In the remainder of the present section we briefly discuss the definition of those alternatives. In the next section, when testing the resulting waiting-time approximations, we will examine the impact of implementing these alternatives.
Denote by $P_{ij}^{(0)}(h, z)$ and $P_{ij}^{(1)}(h, z)$ the conditional probability that $(\mathbf{H}, \mathbf{Z}) = (h, z)$ just after an arrival of S_j at Q_i and after a departure of S_j from Q_i, respectively. These conditional probabilities follow immediately from the dis-

tribution of $(\mathbf{H}, \mathbf{Z})$. Denote by $\mathbf{T}_{ij}^{(0)}(h_j, z_j)$ and $\mathbf{T}_{ij}^{(1)}(h_j, z_j)$ the entrance time into $(\mathbf{H}_j, \mathbf{Z}_j) = (h_j, z_j)$ just after an arrival of S_j at Q_i and after a departure of S_j from Q_i, respectively. The mean values of these entrance times are given by

$$\begin{aligned} \mathrm{E}\mathbf{T}_{ij}^{(b)}(h_j, z_j) &= \frac{r_{ij}s}{1-r_j}(1-b) + \sum_{k=\pi_j^{-1}(i)+1}^{h_j-1} \left(s_{\pi_j(k)} + \frac{r_{\pi_j(k)j}s}{1-r_j} \right) + s_{\pi_j(h_j)}z_j \\ &\approx \frac{\rho_i s}{m-\rho}(1-b) + \sum_{k=\pi_j^{-1}(i)+1}^{h_j-1} \left(s_{\pi_j(k)} + \frac{\rho_{\pi_j(k)}s}{m-\rho} \right) + s_{\pi_j(h_j)}z_j, \end{aligned}$$

$b = 0, 1$, with $k = \pi_j^{-1}(i)$ such that $\pi_j(k) = i$.

For given $(h, z) = (h_1, \ldots, h_m, z_1, \ldots, z_m)$, let $\mathrm{E}\mathbf{T}_{ij_l}^{(b)}(h_{j_l}, z_{j_l})$, $l = 1, \ldots, m$, be the mean entrance times $\mathrm{E}\mathbf{T}_{ij}^{(b)}(h_j, z_j)$, $j = 1, \ldots, m$, ordered in decreasing magnitude. Let $\Delta_{il}^{(b)}(h, z) = \left(\mathrm{E}\mathbf{T}_{ij_{l-1}}^{(b)}(h_{j_{l-1}}, z_{j_{l-1}}) - \mathrm{E}\mathbf{T}_{ij_l}^{(b)}(h_{j_l}, z_{j_l}) \right)$, $l = 1, \ldots, m$, with $\mathrm{E}\mathbf{T}_{ij_0}^{(b)}(h_{j_0}, z_{j_0}) = \mathbf{EC}$, $\mathbf{EC} = s/(m-\rho)$. For given (h, z), $\Delta_{il}^{(b)}(h, z)$ represents the mean of the l-th of the m most recent server interarrival ($b = 0$) or interdeparture ($b = 1$) times at Q_i, $l = 1, \ldots, m$. Denote $\Delta_i^{(b)}(h, z) := \sum_{l=1}^{m} \left(\Delta_{il}^{(b)}(h, z) \right)^2$. The ordinary sum of $\Delta_{il}^{(b)}(h, z)$, $l = 1, \ldots, m$, being always equal to $\mathbf{EC}$, the sum of the *squares* provides a good indication for the *spacing* of the server arrivals at or departures from Q_i. Having this in mind, we define

$$\delta_i^{(b)} := \sum_{j=1}^{m} \sum_{(h,z)} P_{ij}^{(b)}(h, z) \Delta_i^{(b)}(h, z) / (\mathbf{EC})^2 \tag{10.29}$$

for $b = 1$ and $b = 0$ as a measure for the *local* degree of clustering at Q_i under the exhaustive and gated service discipline, respectively. If the degree of clustering at Q_i is high, then for the states (h, z) with large $P_{ij}^{(b)}(h, z)$, one of the $\Delta_{il}^{(b)}(h, z)$'s is approximately equal to $\mathbf{EC}$, while all the other $\Delta_{il}^{(b)}(h, z)$'s are approximately equal to 0, so that $\delta_i^{(b)} \approx m$. On the other hand, if the degree of clustering at Q_i is low, i.e., the $\Delta_{il}^{(b)}(h, z)$'s are the distances between approximately homogeneously distributed points on $[0, \mathbf{EC}]$, then $\delta_i^{(b)} \approx m\kappa$, with $\kappa = \int_{1=x_0 \geq \ldots \geq x_m = 0} \sum_{l=1}^{m} (x_{l-1} - x_l)^2 dx_1 \ldots dx_{m-1}$. If the servers even tend to repel each other, i.e, all the $\Delta_{il}^{(b)}(h, z)$'s are approximately equal to $\mathbf{EC}/m$, then $\delta_i^{(b)} \approx 1$.

We may also refine the measure (10.16) for the *global* degree of clustering. In the spirit of (10.29), we define

$$\delta := \frac{\sum_{(h,z)} \left(\Pr\{(\mathbf{H},\mathbf{Z}) = (h,z)\} - \prod_{l=1}^{m} P_l(h_l,z_l) \right) \sum_{j=1}^{m} \Delta^{(z_j)}_{\pi_j(h_j)}(h,z)}{m(\mathbf{EC})^2 - \sum_{(h,z)} \left(\prod_{l=1}^{m} P_l(h_l,z_l) \right) \sum_{j=1}^{m} \Delta^{(z_j)}_{\pi_j(h_j)}(h,z)}, \quad (10.30)$$

with $P_l(h_l,z_l) = \Pr\{(\mathbf{H}_l,\mathbf{Z}_l) = (h_l,z_l)\}$.
Note that for a high degree of clustering, i.e., for the states (h,z) with large $\Pr\{(\mathbf{H},\mathbf{Z}) = (h,z)\}$, $\Delta^{(z_j)}_{\pi_j(h_j)}(h,z) \approx 1$, $\delta \uparrow 1$. On the other hand, for a low degree of clustering, i.e., $\Pr\{(\mathbf{H},\mathbf{Z}) = (h,z)\} \approx \prod_{l=1}^{m} \Pr\{(\mathbf{H}_l,\mathbf{Z}_l) = (h_l,z_l)\}$, $\delta \downarrow 0$.

10.7 Numerical results

We have tested the waiting-time approximations for a wide range of parameter combinations by comparison with either simulation results or exact numerical results obtained from the power-series algorithm (PSA), as elaborated upon in [150]. The large class of models considered in this chapter has forced us to limit the examples that are used to demonstrate the accuracy of the approximations. Thus we have restricted ourselves to four-queue models with two servers and with exponentially distributed service and switch-over times. Numerous numerical experiments have indicated that for these models the accuracy of the approximations is acceptable, even for rather asymmetric and heavily-loaded systems. In particular, the approximations rightly capture the clustering effects of the visit order, whereas most of the existing approximations completely ignore the considerable influence of the visit order on the waiting times.
The results of the numerical experiments are summarized below. Limited numerical experience (which is however not reported in any further detail below) suggests that the approximations perform similarly for non-exponentially distributed service and switch-over times). We reemphasize that the models considered here are very complex, containing single-server polling models and ordinary multiple-server models as special cases, while the visit order constitutes an additional complicating factor. The accuracy of the approximations should be judged from this perspective.

In order to test the accuracy of the approximations for a wide variety of models, we have considered various variants of a set of models in which the ratios between the arrival rates, (λ_1: λ_2: λ_3: λ_4) and the mean service times (β_1, β_2, β_3, β_4), respectively, are given as follows:

I. (1: 1: 1: 1); (1.0, 1.0, 1.0, 1.0);
II. (1: 1: 3: 3); (1.0, 1.0, 1.0, 1.0);
III. (2: 2: 5: 5); (1.0, 1.0, 0.4, 0.4);
IV. (1: 1: 1: 1); (0.5, 0.5, 1.5, 1.5);
V. (1: 1: 3: 3); (0.5, 1.5, 0.5, 1.5);
VI. (1: 1: 9: 1); (0.5, 0.5, 0.5, 2.5).

By convention, the queues are numbered such that π_1, the visit order of S_1, is always $(1,2,3,4)$. The visit order of S_2 is considered for the cases $\pi_2 = (1,2,3,4)$, $\pi_2 = (1,2,4,3)$, and $\pi_2 = (1,4,3,2)$. For each of the models, all switch-over times are assumed to have mean $s/n = s/4$ with either $s = 0$ or $s = 1$. (In evaluating the approximations we actually took $s = 10^{-6}$, as the formal definition of $(\mathbf{H}, \mathbf{Z})$ is restricted to the case of non-zero switch-over times.) The value of the total load is either $\rho = 0.8$, $\rho = 1.6$, or $\rho = 1.8$. In all considered cases we assume $m_i = m = 2$.

For the models listed above, Tables 10.1.A to 10.6.B show the results for exhaustive service at each of the queues. Tables 10.7.A to 10.8.B show the results for the models II and V with gated service. The rows indicated by 'α' contain the approximations obtained with $\alpha_i = \rho_i/q_i$ as a measure for the local degree of clustering at Q_i, and α as in (10.16) as a measure for the global degree of clustering. The rows marked with 'δ' give the approximations with α_i replaced by $\delta_i^{(1)}$ as in (10.29) for exhaustive service or $\delta_i^{(0)}$ for gated service, and α replaced by δ as in (10.30). The rows indicated by 'exact' contain the 'exact' mean waiting times obtained from either the PSA (for exhaustive service) or simulation (for gated service). (We implemented the PSA only for Bernoulli service, including exhaustive service as special case, but in principle the method may also be developed to compute the mean waiting times for gated service.)

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4) = \rho(0.25,0.25,0.25,0.25)$; $(\beta_1,\beta_2,\beta_3,\beta_4) = (1.0,1.0,1.0,1.0)$; $s = 0.0$.				
		$(\mathbf{EW}_1, \mathbf{EW}_2, \mathbf{EW}_3, \mathbf{EW}_4)$		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.19, 0.19, 0.19, 0.19)	(0.19, 0.19, 0.19, 0.19)	(0.19, 0.19, 0.19, 0.19)
	α	(0.19, 0.19, 0.19, 0.19)	(0.19, 0.19, 0.19, 0.19)	(0.19, 0.19, 0.19, 0.19)
	δ	(0.19, 0.19, 0.19, 0.19)	(0.19, 0.19, 0.19, 0.19)	(0.19, 0.19, 0.19, 0.19)
1.6	exact	(1.78, 1.78, 1.78, 1.78)	(1.78, 1.85, 1.74, 1.74)	(1.78, 1.78, 1.78, 1.78)
	α	(1.78, 1.78, 1.78, 1.78)	(1.76, 1.93, 1.71, 1.71)	(1.78, 1.78, 1.78, 1.78)
	δ	(1.78, 1.78, 1.78, 1.78)	(1.78, 1.92, 1.71, 1.71)	(1.78, 1.78, 1.78, 1.78)
1.8	exact	(4.26, 4.26, 4.26, 4.26)	(4.41, 4.66, 3.98, 3.98)	(4.26, 4.26, 4.26, 4.26)
	α	(4.26, 4.26, 4.26, 4.26)	(4.19, 4.77, 4.04, 4.04)	(4.26, 4.26, 4.26, 4.26)
	δ	(4.26, 4.26, 4.26, 4.26)	(4.25, 4.71, 4.04, 4.04)	(4.26, 4.26, 4.26, 4.26)

TABLE 10.1.A. The mean waiting times for Model I with $s = 0.0$; exhaustive service.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4) = \rho(0.25,0.25,0.25,0.25)$; $(\beta_1,\beta_2,\beta_3,\beta_4) = (1.0,1.0,1.0,1.0)$; $s = 1.0$.				
		$(\mathbf{EW}_1, \mathbf{EW}_2, \mathbf{EW}_3, \mathbf{EW}_4)$		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.77, 0.77, 0.77, 0.77)	(0.77, 0.77, 0.77, 0.77)	(0.76, 0.76, 0.76, 0.76)
	α	(0.78, 0.78, 0.78, 0.78)	(0.78, 0.78, 0.77, 0.77)	(0.78, 0.78, 0.78, 0.78)
	δ	(0.78, 0.78, 0.78, 0.78)	(0.77, 0.78, 0.77, 0.77)	(0.77, 0.77, 0.77, 0.77)
1.6	exact	(3.29, 3.29, 3.29, 3.29)	(3.24, 3.39, 3.09, 3.09)	(3.16, 3.16, 3.16, 3.16)
	α	(3.36, 3.36, 3.36, 3.36)	(3.14, 3.43, 3.05, 3.05)	(3.12, 3.12, 3.12, 3.12)
	δ	(3.52, 3.52, 3.52, 3.52)	(3.24, 3.49, 3.11, 3.11)	(3.14, 3.14, 3.14, 3.14)
1.8	exact	(7.55, 7.55, 7.55, 7.55)	(7.52, 7.86, 6.38, 6.38)	(6.87, 6.87, 6.87, 6.87)
	α	(7.58, 7.58, 7.58, 7.58)	(6.79, 7.72, 6.54, 6.54)	(6.75, 6.75, 6.75, 6.75)
	δ	(7.87, 7.87, 7.87, 7.87)	(7.09, 7.85, 6.74, 6.74)	(6.83, 6.83, 6.83, 6.83)

TABLE 10.1.B. The mean waiting times for Model I with $s = 1.0$; exhaustive service.

Tables 10.1.A and 10.1.B show that the approximations for Model I are very accurate. In particular Table 10.1.A confirms that for completely symmetric systems (including 'symmetric' visit order combinations, i.e., $\pi_2 = (1,2,3,4)$ or $\pi_2 = (1,4,3,2)$), the approximations are exact for exponentially distributed service times and zero switch-over times.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4) = \rho(0.125,0.125,0.375,0.375)$; $(\beta_1,\beta_2,\beta_3,\beta_4) = (1.0,1.0,1.0,1.0)$; $s = 0.0$.				
		(EW_1, EW_2, EW_3, EW_4)		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.21, 0.21, 0.18, 0.18)	(0.21, 0.21, 0.18, 0.18)	(0.21, 0.21, 0.18, 0.18)
	α	(0.20, 0.20, 0.19, 0.19)	(0.20, 0.20, 0.19, 0.19)	(0.20, 0.20, 0.19, 0.19)
	δ	(0.21, 0.21, 0.18, 0.19)	(0.21, 0.21, 0.18, 0.18)	(0.21, 0.21, 0.18, 0.18)
1.6	exact	(2.24, 2.30, 1.63, 1.60)	(2.09, 2.26, 1.65, 1.65)	(2.17, 2.17, 1.65, 1.65)
	α	(2.03, 1.92, 1.67, 1.76)	(1.99, 2.06, 1.70, 1.70)	(2.00, 2.00, 1.70, 1.70)
	δ	(2.18, 2.13, 1.63, 1.68)	(2.05, 2.19, 1.66, 1.66)	(2.08, 2.08, 1.68, 1.68)
1.8	exact	(5.51, 5.92, 3.90, 3.75)	(4.82, 5.36, 4.00, 4.00)	(5.11, 5.11, 4.00, 4.00)
	α	(5.06, 4.83, 3.96, 4.12)	(4.83, 5.14, 4.02, 4.02)	(4.86, 4.86, 4.06, 4.06)
	δ	(5.41, 5.37, 3.86, 3.92)	(5.00, 5.43, 3.95, 3.95)	(5.04, 5.04, 4.00, 4.00)

TABLE 10.2.A. The mean waiting times for Model II with $s = 0.0$; exhaustive service.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4) = \rho(0.125,0.125,0.375,0.375)$; $(\beta_1,\beta_2,\beta_3,\beta_4) = (1.0,1.0,1.0,1.0)$; $s = 1.0$.				
		(EW_1, EW_2, EW_3, EW_4)		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.83, 0.82, 0.74, 0.74)	(0.81, 0.81, 0.73, 0.73)	(0.81, 0.81, 0.74, 0.74)
	α	(0.82, 0.81, 0.75, 0.76)	(0.81, 0.81, 0.74, 0.74)	(0.81, 0.81, 0.75, 0.75)
	δ	(0.86, 0.85, 0.74, 0.75)	(0.83, 0.84, 0.73, 0.73)	(0.83, 0.83, 0.73, 0.73)
1.6	exact	(3.98, 4.04, 2.92, 2.88)	(3.58, 3.86, 2.88, 2.88)	(3.69, 3.69, 2.89, 2.89)
	α	(3.72, 3.53, 3.56, 3.23)	(3.42, 3.55, 2.92, 2.92)	(3.43, 3.43, 2.93, 2.93)
	δ	(4.23, 4.14, 3.16, 3.26)	(3.63, 3.87, 2.94, 2.94)	(3.65, 3.65, 2.95, 2.95)
1.8	exact	(9.04, 9.69, 6.53, 6.32)	(7.27, 8.04, 6.44, 6.44)	(7.73, 7.73, 6.45, 6.45)
	α	(8.69, 8.29, 6.79, 7.07)	(7.53, 8.01, 6.27, 6.27)	(7.54, 7.54, 6.30, 6.30)
	δ	(9.60, 9.43, 6.96, 7.12)	(8.05, 8.74, 6.36, 6.36)	(8.06, 8.06, 6.40, 6.40)

TABLE 10.2.B. The mean waiting times for Model II with $s = 1.0$; exhaustive service.

Tables 10.2.A and 10.2.B show that the results are still accurate when the arrival rates are fairly asymmetric, even for heavily-loaded systems.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4) = \rho(0.25,0.25,0.625,0.625)$; $(\beta_1,\beta_2,\beta_3,\beta_4) = (1.0,1.0,0.4,0.4)$; $s = 0.0$.				
		(EW_1, EW_2, EW_3, EW_4)		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.13, 0.12, 0.13, 0.13)	(0.13, 0.12, 0.13, 0.13)	(0.13, 0.13, 0.13, 0.13)
	α	(0.13, 0.13, 0.13, 0.13)	(0.13, 0.14, 0.13, 0.13)	(0.13, 0.13, 0.13, 0.13)
	δ	(0.13, 0.13, 0.13, 0.13)	(0.13, 0.14, 0.13, 0.13)	(0.13, 0.13, 0.13, 0.13)
1.6	exact	(1.25, 1.14, 1.22, 1.34)	(1.27, 1.20, 1.24, 1.24)	(1.21, 1.21, 1.26, 1.26)
	α	(1.24, 1.24, 1.24, 1.24)	(1.23, 1.35, 1.20, 1.20)	(1.24, 1.24, 1.24, 1.24)
	δ	(1.24, 1.24, 1.24, 1.24)	(1.25, 1.34, 1.20, 1.20)	(1.24, 1.24, 1.24, 1.24)
1.8	exact	(3.01, 2.75, 2.99, 3.22)	(3.15, 3.14, 2.83, 2.83)	(2.94, 2.94, 3.01, 3.01)
	α	(2.98, 2.98, 2.98, 2.98)	(2.94, 3.34, 2.83, 2.83)	(2.98, 2.98, 2.98, 2.98)
	δ	(2.98, 2.98, 2.98, 2.98)	(2.98, 3.30, 2.83, 2.83)	(2.98, 2.98, 2.98, 2.98)

TABLE 10.3.A. The mean waiting times for Model III with $s = 0.0$; exhaustive service.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4) = \rho(0.25,0.25,0.625,0.625)$; $(\beta_1,\beta_2,\beta_3,\beta_4) = (1.0,1.0,0.4,0.4)$; $s = 1.0$.				
		$(\mathbf{EW}_1,\mathbf{EW}_2,\mathbf{EW}_3,\mathbf{EW}_4)$		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.72, 0.70, 0.70, 0.72)	(0.71, 0.70, 0.70, 0.70)	(0.70, 0.70, 0.70, 0.70)
	α	(0.72, 0.72, 0.72, 0.72)	(0.72, 0.73, 0.72, 0.72)	(0.71, 0.71, 0.71, 0.71)
	δ	(0.72, 0.72, 0.72, 0.72)	(0.71, 0.72, 0.71, 0.71)	(0.71, 0.71, 0.71, 0.71)
1.6	exact	(2.80, 2.60, 2.79, 2.99)	(2.76, 2.72, 2.60, 2.60)	(2.59, 2.59, 2.65, 2.65)
	α	(2.83, 2.83, 2.83, 2.83)	(2.61, 2.85, 2.54, 2.54)	(2.59, 2.59, 2.59, 2.59)
	δ	(2.98, 2.98, 2.98, 2.98)	(2.71, 2.92, 2.60, 2.60)	(2.61, 2.61, 2.61, 2.61)
1.8	exact	(6.43, 5.97, 6.38, 6.70)	(6.42, 6.60, 5.24, 5.24)	(5.55, 5.55, 5.63, 5.63)
	α	(6.30, 6.30, 6.30, 6.30)	(5.53, 6.29, 5.33, 5.33)	(5.47, 5.47, 5.47, 5.47)
	δ	(6.59, 6.59, 6.59, 6.59)	(5.81, 6.43, 5.52, 5.52)	(5.55, 5.55, 5.55, 5.55)

TABLE 10.3.B. The mean waiting times for Model III with $s = 1.0$; exhaustive service.

In the cases considered in Tables 10.3.A and 10.3.B the arrival rates and the service rates are rather asymmetric, but the load offered to each of the queues is the same. By construction, the approximated ratios of the mean waiting times only depend on the λ_i's and β_i's through the ρ_i's. As the ρ_i's are all equal here, the approximated mean waiting times are also all equal for 'symmetric' visit order combinations, i.e., $\pi_2 = (1,2,3,4)$ or $\pi_2 = (1,4,3,2)$. The numerical results show that the *true* ratios of the mean waiting times *do* depend on the individual λ_i's and β_i's, but that the accuracy of the approximated mean waiting times is still acceptable.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4) = \rho(0.25,0.25,0.25,0.25)$; $(\beta_1,\beta_2,\beta_3,\beta_4) = (0.5,0.5,1.5,1.5)$; $s = 0.0$.				
		$(\mathbf{EW}_1,\mathbf{EW}_2,\mathbf{EW}_3,\mathbf{EW}_4)$		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.25, 0.27, 0.23, 0.21)	(0.25, 0.27, 0.22, 0.22)	(0.26, 0.26, 0.22, 0.22)
	α	(0.25, 0.25, 0.23, 0.24)	(0.25, 0.26, 0.23, 0.23)	(0.25, 0.25, 0.23, 0.23)
	δ	(0.26, 0.26, 0.23, 0.23)	(0.26, 0.26, 0.23, 0.23)	(0.26, 0.26, 0.23, 0.23)
1.6	exact	(2.79, 3.07, 2.05, 1.88)	(2.55, 2.92, 2.03, 2.03)	(2.74, 2.74, 2.03, 2.03)
	α	(2.53, 2.40, 2.08, 2.20)	(2.48, 2.57, 2.12, 2.12)	(2.50, 2.50, 2.13, 2.13)
	δ	(2.72, 2.66, 2.03, 2.10)	(2.57, 2.74, 2.08, 2.08)	(2.60, 2.60, 2.10, 2.10)
1.8	exact	(7.00, 7.58, 4.84, 4.49)	(5.90, 6.77, 4.99, 4.99)	(6.38, 6.38, 4.96, 4.96)
	α	(6.33, 6.03, 4.95, 5.15)	(6.04, 6.43, 5.03, 5.03)	(6.08, 6.08, 5.08, 5.08)
	δ	(6.68, 6.56, 4.84, 4.96)	(6.25, 6.78, 4.93, 4.93)	(6.30, 6.30, 5.00, 5.00)

TABLE 10.4.A The mean waiting times for Model IV with $s = 0.0$; exhaustive service.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4) = \rho(0.25,0.25,0.25,0.25)$; $(\beta_1,\beta_2,\beta_3,\beta_4) = (0.5,0.5,1.5,1.5)$; $s = 1.0$.				
		$(\mathbf{EW}_1,\mathbf{EW}_2,\mathbf{EW}_3,\mathbf{EW}_4)$		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.87, 0.88, 0.79, 0.78)	(0.85, 0.88, 0.78, 0.78)	(0.86, 0.86, 0.78, 0.78)
	α	(0.87, 0.86, 0.79, 0.81)	(0.86, 0.87, 0.79, 0.79)	(0.86, 0.86, 0.79, 0.79)
	δ	(0.91, 0.90, 0.78, 0.80)	(0.88, 0.89, 0.78, 0.78)	(0.89, 0.89, 0.78, 0.78)
1.6	exact	(4.56, 4.91, 3.35, 3.13)	(4.05, 4.53, 3.28, 3.28)	(4.28, 4.28, 3.28, 3.28)
	α	(4.23, 4.01, 3.47, 3.67)	(3.92, 4.06, 3.35, 3.35)	(3.93, 3.93, 3.36, 3.36)
	δ	(4.78, 4.67, 3.56, 3.68)	(4.14, 4.42, 3.35, 3.35)	(4.17, 4.17, 3.37, 3.37)
1.8	exact	(10.55, 11.20, 7.23, 7.00)	(8.34, 9.51, 7.41, 7.41)	(9.04, 9.04, 7.39, 7.39)
	α	(9.95, 9.49, 7.78, 8.10)	(8.74, 9.30, 7.27, 7.27)	(8.75, 8.75, 7.32, 7.32)
	δ	(10.93, 10.75, 7.92, 8.11)	(9.30, 10.10, 7.35, 7.35)	(9.32, 9.32, 7.40, 7.40)

TABLE 10.4.B. The mean waiting times for Model IV with $s = 1.0$; exhaustive service.

In Model IV the service times are asymmetric, whereas the arrival rates are the same. Tables 10.4.A and 10.4.B show that the accuracy of the results is

acceptable, even in heavily-loaded systems.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4)=\rho(0.125,0.125,0.375,0.375)$; $(\beta_1,\beta_2,\beta_3,\beta_4)=(0.5,1.5,0.5,1.5)$; $s=0.0$.				
		(EW_1, EW_2, EW_3, EW_4)		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.27, 0.25, 0.26, 0.21)	(0.28, 0.26, 0.26, 0.21)	(0.27, 0.25, 0.26, 0.21)
	α	(0.27, 0.25, 0.25, 0.23)	(0.27, 0.25, 0.25, 0.23)	(0.26, 0.26, 0.25, 0.23)
	δ	(0.29, 0.26, 0.26, 0.22)	(0.28, 0.26, 0.26, 0.22)	(0.27, 0.27, 0.26, 0.22)
1.6	exact	(3.24, 2.65, 2.99, 1.67)	(3.13, 2.84, 2.76, 1.71)	(3.02, 2.81, 2.78, 1.72)
	α	(2.75, 2.63, 2.61, 1.90)	(2.72, 2.65, 2.48, 1.94)	(2.59, 2.72, 2.42, 1.95)
	δ	(3.20, 2.75, 2.76, 1.76)	(3.08, 2.82, 2.54, 1.82)	(2.65, 2.91, 2.50, 1.85)
1.8	exact	(8.26, 6.68, 7.53, 3.73)	(7.51, 7.17, 6.59, 4.03)	(7.12, 7.14, 6.61, 4.06)
	α	(7.01, 6.64, 6.62, 4.28)	(6.77, 6.74, 6.18, 4.42)	(6.36, 6.87, 6.04, 4.46)
	δ	(7.97, 6.84, 6.86, 4.02)	(7.74, 7.08, 6.34, 4.14)	(6.50, 7.29, 6.26, 4.24)

TABLE 10.5.A. The mean waiting times for Model V with $s = 0.0$; exhaustive service.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4)=\rho(0.125,0.125,0.375,0.375)$; $(\beta_1,\beta_2,\beta_3,\beta_4)=(0.5,1.5,0.5,1.5)$; $s=1.0$.				
		(EW_1, EW_2, EW_3, EW_4)		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.91, 0.85, 0.86, 0.74)	(0.90, 0.86, 0.85, 0.74)	(0.89, 0.86, 0.85, 0.74)
	α	(0.89, 0.85, 0.85, 0.75)	(0.88, 0.85, 0.84, 0.75)	(0.88, 0.85, 0.84, 0.75)
	δ	(0.98, 0.89, 0.90, 0.74)	(0.94, 0.88, 0.86, 0.73)	(0.91, 0.89, 0.85, 0.73)
1.6	exact	(5.12, 4.28, 4.68, 2.69)	(4.82, 4.42, 4.19, 2.74)	(4.58, 4.41, 4.22, 2.76)
	α	(4.43, 4.23, 4.20, 3.06)	(4.19, 4.08, 3.81, 2.98)	(3.99, 4.19, 3.73, 3.00)
	δ	(5.43, 4.67, 4.69, 2.98)	(4.88, 4.47, 4.03, 2.89)	(4.19, 4.60, 3.95, 2.93)
1.8	exact	(11.94, 10.95, 11.09, 5.44)	(10.56, 10.29, 9.15, 5.89)	(9.88, 10.49, 9.27, 5.89)
	α	(10.59, 10.03, 10.01, 6.47)	(9.53, 9.48, 8.69, 6.22)	(8.95, 9.67, 8.50, 6.28)
	δ	(12.56, 10.79, 10.81, 6.34)	(11.28, 10.32, 9.24, 6.03)	(9.43, 10.57, 9.08, 6.15)

TABLE 10.5.B. The mean waiting times for Model V with $s = 1.0$; exhaustive service.

In the models considered in Tables 10.5.A and 10.5.B the arrival rates as well as the service times are asymmetric. It is shown that in these cases the approximations are less accurate than in the cases considered above, but still acceptable. In Models I-IV both approximations yielded similar results, but here the δ-approximation tends to outperform the α-approximation. Apparently the latter fails to detect the clustering at the lightly-loaded queues that are visited after the heavily-loaded Q_4.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4)=\rho(0.125,0.125,1.125,0.125)$; $(\beta_1,\beta_2,\beta_3,\beta_4)=(0.5,0.5,0.5,2.5)$; $s=0.0$.				
		(EW_1, EW_2, EW_3, EW_4)		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.21, 0.22, 0.18, 0.19)	(0.23, 0.23, 0.18, 0.19)	(0.23, 0.23, 0.18, 0.19)
	α	(0.24, 0.24, 0.20, 0.23)	(0.24, 0.24, 0.21, 0.22)	(0.24, 0.24, 0.21, 0.22)
	δ	(0.26, 0.26, 0.20, 0.23)	(0.26, 0.26, 0.20, 0.22)	(0.26, 0.25, 0.20, 0.22)
1.6	exact	(2.96, 3.17, 1.65, 2.04)	(2.99, 3.25, 1.62, 2.08)	(3.13, 3.12, 1.62, 2.08)
	α	(2.45, 2.31, 1.70, 2.39)	(2.43, 2.47, 1.77, 2.23)	(2.48, 2.41, 1.77, 2.23)
	δ	(2.94, 2.88, 1.61, 2.34)	(2.60, 2.75, 1.70, 2.28)	(2.76, 2.54, 1.70, 2.28)
1.8	exact	(7.94, 8.50, 3.78, 4.91)	(7.54, 8.18, 3.83, 5.05)	(8.05, 7.75, 3.84, 5.07)
	α	(9.14, 8.57, 5.21, 8.22)	(8.29, 8.55, 5.46, 7.92)	(8.50, 8.18, 5.47, 7.93)
	δ	(10.44, 10.30, 5.00, 8.01)	(8.88, 9.43, 5.21, 8.09)	(9.39, 8.66, 5.23, 8.11)

TABLE 10.6.A. The mean waiting times for Model VI with $s = 0.0$; exhaustive service.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4) = \rho(0.125, 0.125, 1.125, 0.125)$; $(\beta_1,\beta_2,\beta_3,\beta_4) = (0.5, 0.5, 0.5, 2.5)$; $s = 1.0$.				
		$(\mathbf{EW}_1, \mathbf{EW}_2, \mathbf{EW}_3, \mathbf{EW}_4)$		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.84, 0.84, 0.69, 0.79)	(0.86, 0.86, 0.69, 0.77)	(0.86, 0.85, 0.69, 0.77)
	α	(0.86, 0.85, 0.72, 0.81)	(0.86, 0.86, 0.73, 0.79)	(0.86, 0.86, 0.73, 0.79)
	δ	(0.95, 0.94, 0.72, 0.84)	(0.92, 0.92, 0.72, 0.80)	(0.93, 0.90, 0.72, 0.80)
1.6	exact	(4.69, 5.05, 2.67, 3.35)	(4.63, 4.95, 2.62, 3.36)	(4.93, 4.86, 2.62, 3.34)
	α	(4.05, 3.81, 2.80, 3.95)	(3.85, 3.90, 2.80, 3.52)	(3.93, 3.81, 2.80, 3.52)
	δ	(5.19, 5.09, 2.84, 4.13)	(4.29, 4.52, 2.80, 3.75)	(4.55, 4.18, 2.80, 3.75)
1.8	exact	(11.44, 12.45, 5.63, 5.70)	(9.99, 11.10, 5.70, 7.36)	(10.96, 9.97, 5.74, 7.36)
	α	(13.91, 13.04, 7.93, 12.50)	(11.71, 12.08, 7.71, 11.18)	(11.99, 11.53, 7.72, 11.18)
	δ	(16.66, 16.43, 7.98, 12.78)	(13.19, 14.02, 7.75, 12.03)	(13.94, 12.85, 7.76, 12.04)

TABLE 10.6.B. The mean waiting times for Model VI with $s = 1.0$; exhaustive service.

Model VI is a typical example of a very asymmetric system. In such cases, the accuracy of all waiting-time approximations in the literature degrades significantly, even in single-server systems. Tables 10.6.A and 10.6.B show that the accuracy of the waiting-time approximation presented in this chapter also degrades when the model is very asymmetric, but remains acceptable as long as the load is not too high.

We now check the accuracy of the approximation for multiple-server systems with gated service at all queues.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4) = \rho(0.125, 0.125, 0.375, 0.375)$; $(\beta_1,\beta_2,\beta_3,\beta_4) = (1.0, 1.0, 1.0, 1.0)$; $s = 0.0$.				
		$(\mathbf{EW}_1, \mathbf{EW}_2, \mathbf{EW}_3, \mathbf{EW}_4)$		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.18, 0.17, 0.19, 0.20)	(0.18, 0.17, 0.19, 0.20)	(0.18, 0.18, 0.19, 0.19)
	α	(0.16, 0.16, 0.20, 0.20)	(0.16, 0.16, 0.20, 0.20)	(0.16, 0.16, 0.20, 0.20)
	δ	(0.17, 0.17, 0.20, 0.20)	(0.17, 0.17, 0.20, 0.20)	(0.17, 0.17, 0.20, 0.20)
1.6	exact	(1.54, 1.52, 1.82, 1.85)	(1.46, 1.51, 1.85, 1.86)	(1.46, 1.46, 1.86, 1.86)
	α	(1.38, 1.33, 1.90, 1.94)	(1.29, 1.32, 1.94, 1.94)	(1.29, 1.29, 1.94, 1.94)
	δ	(1.56, 1.54, 1.84, 1.87)	(1.44, 1.53, 1.88, 1.88)	(1.44, 1.44, 1.89, 1.89)
1.8	exact	(3.60, 3.59, 4.34, 4.40)	(3.22, 3.42, 4.46, 4.46)	(3.29, 3.29, 4.47, 4.47)
	α	(3.40, 3.28, 4.53, 4.61)	(3.03, 3.16, 4.65, 4.65)	(3.03, 3.03, 4.67, 4.67)
	δ	(3.72, 3.67, 4.43, 4.48)	(3.36, 3.63, 4.52, 4.52)	(3.36, 3.36, 4.56, 4.56)

TABLE 10.7.A. The mean waiting times for Model II with $s = 0.0$; gated service.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4) = \rho(0.125, 0.125, 0.375, 0.375)$; $(\beta_1,\beta_2,\beta_3,\beta_4) = (1.0, 1.0, 1.0, 1.0)$; $s = 1.0$.				
		$(\mathbf{EW}_1, \mathbf{EW}_2, \mathbf{EW}_3, \mathbf{EW}_4)$		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.84, 0.83, 0.91, 0.92)	(0.83, 0.83, 0.91, 0.91)	(0.83, 0.83, 0.91, 0.91)
	α	(0.75, 0.75, 0.95, 0.96)	(0.75, 0.75, 0.95, 0.95)	(0.75, 0.75, 0.95, 0.95)
	δ	(0.83, 0.82, 0.94, 0.95)	(0.80, 0.81, 0.93, 0.93)	(0.80, 0.80, 0.93, 0.93)
1.6	exact	(3.85, 3.79, 4.56, 4.61)	(3.27, 3.44, 4.28, 4.28)	(3.35, 3.35, 4.35, 4.35)
	α	(3.25, 3.12, 4.47, 4.58)	(2.80, 2.88, 4.21, 4.21)	(2.80, 2.80, 4.21, 4.21)
	δ	(3.95, 3.88, 4.66, 4.73)	(3.22, 3.43, 4.22, 4.22)	(3.22, 3.22, 4.21, 4.21)
1.8	exact	(8.38, 8.31, 10.15, 10.22)	(6.47, 6.88, 9.33, 9.37)	(6.94, 6.94, 9.84, 9.84)
	α	(7.69, 7.42, 10.23, 10.42)	(6.00, 6.26, 9.22, 9.22)	(5.96, 5.96, 9.19, 9.19)
	δ	(8.91, 8.81, 10.62, 10.74)	(6.96, 7.53, 9.37, 9.37)	(6.89, 6.89, 9.37, 9.37)

TABLE 10.7.B. The mean waiting times for Model II with $s = 1.0$; gated service.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4) = \rho(0.125, 0.125, 0.375, 0.375)$; $(\beta_1,\beta_2,\beta_3,\beta_4) = (0.5, 1.5, 0.5, 1.5)$; $s = 0.0$.				
		(EW_1, EW_2, EW_3, EW_4)		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.20, 0.21, 0.21, 0.25)	(0.20, 0.22, 0.21, 0.24)	(0.20, 0.21, 0.21, 0.24)
	α	(0.17, 0.20, 0.20, 0.27)	(0.17, 0.20, 0.19, 0.27)	(0.17, 0.21, 0.19, 0.27)
	δ	(0.20, 0.21, 0.22, 0.26)	(0.20, 0.22, 0.21, 0.26)	(0.19, 0.22, 0.21, 0.26)
1.6	exact	(1.71, 1.91, 1.86, 2.46)	(1.68, 1.94, 1.81, 2.52)	(1.54, 1.91, 1.80, 2.49)
	α	(1.39, 1.74, 1.73, 2.64)	(1.28, 1.68, 1.61, 2.71)	(1.23, 1.72, 1.60, 2.71)
	δ	(1.77, 1.93, 1.93, 2.47)	(1.64, 1.91, 1.74, 2.55)	(1.40, 1.95, 1.72, 2.57)
1.8	exact	(3.91, 4.45, 4.36, 5.87)	(3.76, 4.54, 4.04, 6.18)	(3.35, 4.62, 4.11, 6.37)
	α	(3.46, 4.28, 4.28, 6.24)	(2.96, 4.04, 3.84, 6.52)	(2.83, 4.10, 3.80, 6.52)
	δ	(4.19, 4.60, 4.59, 5.95)	(3.85, 4.55, 4.12, 6.16)	(3.21, 4.62, 4.08, 6.22)

TABLE 10.8.A. The mean waiting times for Model V with $s = 0.0$; gated service.

$(\lambda_1,\lambda_2,\lambda_3,\lambda_4) = \rho(0.125, 0.125, 0.375, 0.375)$; $(\beta_1,\beta_2,\beta_3,\beta_4) = (0.5, 1.5, 0.5, 1.5)$; $s = 1.0$.				
		(EW_1, EW_2, EW_3, EW_4)		
ρ	π_2	(1, 2, 3, 4)	(1, 2, 4, 3)	(1, 4, 3, 2)
0.8	exact	(0.85, 0.87, 0.89, 0.99)	(0.85, 0.87, 0.88, 0.98)	(0.84, 0.88, 0.90, 1.00)
	α	(0.70, 0.80, 0.80, 1.12)	(0.69, 0.80, 0.79, 1.11)	(0.69, 0.80, 0.79, 1.11)
	δ	(0.84, 0.89, 0.90, 1.07)	(0.81, 0.88, 0.85, 1.06)	(0.78, 0.88, 0.85, 1.06)
1.6	exact	(3.75, 4.18, 4.14, 5.42)	(3.51, 4.11, 3.73, 5.39)	(3.16, 4.09, 3.76, 5.40)
	α	(2.95, 3.69, 3.68, 5.61)	(2.54, 3.36, 3.22, 5.40)	(2.46, 3.43, 3.19, 5.41)
	δ	(4.06, 4.42, 4.42, 5.65)	(3.41, 3.98, 3.62, 5.32)	(2.91, 4.05, 3.56, 5.33)
1.8	exact	(8.56, 9.66, 9.56, 12.74)	(7.15, 8.84, 7.54, 12.18)	(6.26, 9.10, 7.79, 12.71)
	α	(7.07, 8.76, 8.75, 12.76)	(5.42, 7.41, 7.05, 11.95)	(5.18, 7.52, 6.98, 11.96)
	δ	(9.12, 10.02, 10.00, 12.95)	(7.45, 8.80, 7.97, 11.92)	(6.17, 8.88, 7.84, 11.95)

TABLE 10.8.B. The mean waiting times for Model V with $s = 1.0$; gated service.

Tables 10.7.A to 10.8.B show similar results as for the corresponding models with exhaustive service: the accuracy is acceptable for systems which are not too asymmetric, even for heavily-loaded systems in which the switch-over times are significant.

Discussion of the numerical results

We have tested the accuracy of the waiting-time approximations extensively for a broad set of parameter combinations, viz., for lightly-, medium- and heavily-loaded systems, with symmetric and asymmetric arrival and service rates, with negligible and non-negligible switch-over times, and with varying visit order combinations. In general, the results are fairly accurate. Because the clustering effects of the visit order are explicitly taken into account, the results are still accurate in cases where the server bunching is significant, whereas most of the existing approximations completely ignore the influence of the visit order on the waiting times.

As discussed extensively in Sections 10.4, 10.5, and 10.6, the waiting-time approximations presented in this chapter are based on a series of assumptions, each of which inherently forms a source of inaccuracy. The first source of inaccuracy stems from the estimation of the ratios between the mean waiting times which, in turn, is composed of a number of approximations for (i) the mean waiting times in terms of the mean residual cycle times (cf. (10.9), (10.10)), (ii) the ratios between the mean residual cycle times (cf. (10.11), (10.12)), and (iii) the value of q_i (cf. (10.18)). The second error source is the estimation of

the mean amount of work in the system (cf. (10.17)). Extensive simulation experiments have been performed to check the impact of each of these error sources on the finally observed error in the approximated mean waiting times. Inspection of the numerical results has revealed that the estimation of the mean amount of work in the system, $\mathbf{EV}$, according to (10.17), is rather accurate. The error in the estimation of $\mathbf{EV}$ is typically less than 5% in fairly symmetric systems, even under heavy traffic, and remains well below 10% for rather asymmetric systems, even when the offered load is high.

The main source of inaccuracy stems from the estimation of the ratios between the mean waiting times. The approximation of q_i, i.e., the probability that at least one of the servers is busy at Q_i, is quite accurate in many cases, also when the clustering effect is significant, with errors typically below 10%. We found that q_i is underestimated in most of the cases. This is probably due to the fact that the clustering effect in the approximative approach is somewhat exaggerated because of the assumption that all the visiting servers depart from a queue simultaneously. The approximation of q_i may become inaccurate for very asymmetric systems under a heavy-traffic scenario. Other inaccuracies stem from the approximation of the mean waiting times in terms of the mean residual cycle times (cf. (10.9), (10.10)). For systems with the exhaustive service discipline, the mean waiting times are usually underestimated according to (10.9) (where q_i and $\mathbf{ERD}_i$ are taken to be their respective true (simulated) values), whereas in case of the gated service discipline, the mean waiting times are somewhat overestimated according to (10.10). Apparently, in the case of exhaustive service the approximation (10.9) is too optimistic and, in the case of gated service, the approximation (10.10) is rather pessimistic. However, in both cases the *ratios* between the overestimated mean waiting times appear to be rather robust with respect to these errors, so that the errors resulting from (10.9) and (10.10) only have a marginal impact on the finally obtained waiting-time approximations. The ratios between the mean residual cycle times are estimated by the ratios between the estimated average processing rates according to (10.11) and (10.12). Numerical experience has taught us that the quality of these estimations is quite good, with errors typically up to 10%, except for very asymmetric systems under heavy traffic, in which cases the accuracy of the approximations (10.11) and (10.12) may degrade significantly.

In the general approach developed in Sections 10.4, 10.5, and 10.6 we used $\alpha_i = \rho_i/q_i$ as a measure for the local degree of clustering and α in (10.16) as a measure for the global degree of clustering, where both degrees of clustering are based on estimation of the average processing speed. However, for situations in which the average processing speed does not provide a good indication for the degree of clustering, we defined in the second part of Section 10.6 alternative clustering measures, viz., sum-of-square-like spacing measures for the positions of the servers in the system (cf. (10.29), (10.30)). Comparing the accuracy of the waiting-time approximations based on both clustering measures (in the tables indicated by α and δ) has not indicated a clear superiority of one of the two measures; the α-approximation is 122 out of the 468 times more than 10%

off; the δ-approximation 96 times.
Summarizing, in general the approximations presented in this chapter lead to fairly accurate results when the system load is not too high and the system parameters are not too asymmetric. Apparently, the approximations cover the main characteristics of the extremely complicated behavior of multiple-server polling systems. When the system is very asymmetric and the offered load is very high, the accuracy of the approximations may however degrade significantly.

10.8 Concluding remarks and suggestions for further research

In the present chapter we focused on the case $m_i = m$, i.e., all the m servers may visit Q_i simultaneously. It would be interesting to derive waiting-time approximations for the case $m_i < m$, in particular for $m_i = 1$. In some respects the analysis will be somewhat facilitated then. Formulae (10.9), (10.10) e.g. are exact for $m_i = 1$. Also the probabilities q_i that show up in these formulae are simply known to be ρ_i for $m_i = 1$. In certain other respects the analysis will however be more complicated. The average processing speed $\alpha_i = \rho_i/q_i$ will always be equal to 1 for $m_i = 1$, so that it can no longer be used as a measure for the degree of clustering at Q_i. The more detailed measures $\delta_i^{(b)}$ can still be used. It will however be harder to approximate the simultaneous distribution of $(\mathbf{H}, \mathbf{Z})$ needed to determine these measures, as for $m_i < m$ there are also instantaneous passages through states with more than m_i servers at Q_i. Also the derivation of an approximative pseudo-conservation law will be considerably harder.
In the present chapter we have considered systems in which each of the servers visits the queues according to its own strictly cyclic schedule, and where the switch-over times only depend on the next queue to be visited. It would be interesting to generalize the approximations to systems in which the servers may use a not necessarily cyclic schedule or a random schedule, or where the switch-over times also depend on the previous queue visited.

Bibliography

[1] Ackroyd, M.H. (1985). Numerical computation of delays in clocked schedules. *AT & T Techn. J.* **64**, 617-631.

[2] Ajmone Marsan, M., Donatelli, S., Neri, F. (1990). GSPN models of Markovian multiserver multiqueue systems. *Perf. Eval.* **11**, 227-240.

[3] Ajmone Marsan, M., Donatelli, S., Neri, F. (1991). Multiserver multiqueue systems with limited service and zero walk time. In: *Proc. INFOCOM '91*, 1178-1188.

[4] Ajmone Marsan, M., De Moraes, L.F., Donatelli, S., Neri, F. (1990). Analysis of symmetric nonexhaustive polling with multiple servers. In: *Proc. INFOCOM '90*, 284-295.

[5] Ajmone Marsan, M., De Moraes, L.F., Donatelli, S., Neri, F. (1992). Cycles and waiting times in symmetric exhaustive and gated multiserver multiqueue systems. In: *Proc. INFOCOM '92*, 2315-2324.

[6] Altman, E., Foss, S. (1993). Polling on a graph with general arrival and service time distribution. Report INRIA Sophia Antipolis.

[7] Altman, E., Khamisy A., Yechiali, U. (1992). On elevator polling with globally gated regime. *Queueing Systems* **11**, *Special Issue on Polling Models*, 85-90.

[8] Altman, E., Konstantopoulos, P., Liu, Z. (1992). Stability, monotonicity and invariant quantities in general polling systems. *Queueing Systems* **11**, *Special Issue on Polling Models*, 35-57.

[9] Altman, E., Yechiali, U. (1994). Polling in a closed network. To appear in *Prob. Eng. Inf. Sc.*

[10] Anily, S., Federgruen, A. (1991). Structured partitioning problems. *Oper. Res.* **39**, 130-149.

[11] Arem, B. van (1990). *Queueing Models for Slotted Transmission Systems.* Ph.D. Thesis, Twente University, Enschede.

[12] Arian, Y., Levy, Y. (1992). Algorithms for generalized round robin routing. *Oper. Res. Lett.* **12**, 313-319.

[13] Athreya, K.B., Ney, P.E. (1972). *Branching Processes* (Springer, Berlin).

[14] Avi-Itzhak, B., Maxwell, W.L., Miller, W. (1965). Queuing with alternating priorities. *Oper. Res.* **13**, 306-318.

[15] Baker, J.E., Rubin, I. (1987). Polling with a general-service order table. *IEEE Trans. Commun.* **35**, 283-288.

[16] Bertsekas, D., Gallager, R. (1992). *Data Networks* (Prentice-Hall, Englewood Cliffs, NJ, 2nd ed.).

[17] Bertsimas, D.J., Van Ryzin, G. (1993). Stochastic and dynamic vehicle routing with general demand and interarrival time distributions. *Adv. Appl. Prob.* **25**, 947-978.

[18] Bhuyan, L.N., Ghosal, D., Yang, Q. (1989). Approximate analysis of single and multiple ring networks. *IEEE Trans. Comput.* **38**, 1027-1040.

[19] Bisdikian, C. (1993). The random N-policy. Preprint.

[20] Bisdikian, C., Merakos, L. (1992). Output process from a continuous token-ring local area network. *IEEE Trans. Commun.* **40**, 1796-1799.

[21] Blanc, J.P.C. (1990). A numerical approach to cyclic-service queueing models. *Queueing Systems* **6**, 173-188.

[22] Blanc, J.P.C. (1991). The power-series algorithm applied to cyclic polling systems. *Commun. Stat. - Stoch. Mod.* **7**, 527-545.

[23] Blanc, J.P.C., Van der Mei, R.D. (1992). Optimization of polling systems with Bernoulli schedules. To appear in *Perf. Eval.*

[24] Blanc, J.P.C., Van der Mei, R.D. (1994). The power-series algorithm applied to polling systems with a dormant server. In: *The Fundamental Role of Teletraffic in the Evolution of Telecommunications Networks, Proc. ITC-14*, eds. J. Labetoulle, J.W. Roberts (North-Holland, Amsterdam), 865-874.

[25] Borovkov, A.A., Schassberger, R. (1994). Ergodicity of a polling network. *Stoch. Proc. Appl.* **50**, 253-262.

[26] Borst, S.C. (1993). A polling system with a dormant server. CWI Report BS-R9313.

[27] Borst, S.C. (1994). A pseudo-conservation law for a polling system with a dormant server. In: *The Fundamental Role of Teletraffic in the Evolution of Telecommunications Networks, Proc. ITC-14*, eds. J. Labetoulle, J.W. Roberts (North-Holland, Amsterdam), 729-742.

[28] Borst, S.C. (1993). A globally gated polling system with a dormant server. Submitted to *Prob. Eng. Inf. Sc.*

[29] Borst, S.C. (1994). Polling systems with multiple coupled servers. CWI Report BS-R9408. Submitted to *Queueing Systems.*

[30] Borst, S.C. (1994). Optimal probabilistic allocation of customer types to servers. CWI Report BS-R9415. Submitted to *J. ACM.*

[31] Borst, S.C., Boxma, O.J. (1994). Polling models with and without switchover times. CWI Report BS-R9421. Submitted to *Oper. Res.*

[32] Borst, S.C., Boxma, O.J., Combé, M.B. (1992). Collection of customers: a correlated $M/G/1$ queue. In: *Perf. Eval. Review* **20**, 47-59 (Proc. 1992 ACM SIGMETRICS & Performance '92).

[33] Borst, S.C., Boxma, O.J., Combé, M.B. (1993). An $M/G/1$ queue with customer collection. *Commun. Stat. - Stoch. Mod.* **9**, 341-371.

[34] Borst, S.C., Boxma, O.J., Harink, J.H.A., Huitema, G.B. (1992). Optimization of fixed time polling schemes. CWI Report BS-R9237. To appear in *Telecommunication Systems.*

[35] Borst, S.C., Boxma, O.J., Levy, H. (1993). The use of service limits for efficient operation of multi-station single-medium communication systems. CWI Report BS-R9312. Submitted to *IEEE Trans. Netw.*

[36] Borst, S.C., Van der Mei, R.D. (1994). Waiting-time approximations for multiple-server polling systems. CWI Report BS-R9428.

[37] Boxma, O.J. (1985). Two symmetric queues with alternating service and switching times. In: *Proc. Performance '84*, ed. E. Gelenbe (North-Holland, Amsterdam), 409-431.

[38] Boxma, O.J. (1989). Workloads and waiting times in single-server queues with multiple customer classes. *Queueing Systems* **5**, 185-214.

[39] Boxma, O.J. (1991). Analysis and optimization of polling systems. In: *Queueing, Performance and Control in ATM*, eds. J.W. Cohen, C.D. Pack (North-Holland, Amsterdam), 173-183.

[40] Boxma, O.J., Combé, M.B. (1993). The correlated $M/G/1$ queue. *AEÜ* **47**, *Special Issue on Teletraffic Theory and Engineering in Memory of Félix Pollaczek*, 330-335.

[41] Boxma, O.J., Combé, M.B. (1994). Optimization of static traffic allocation policies. *Theor. Comput. Sc.* **125**, 17-43.

[42] Boxma, O.J., Groenendijk, W.P. (1987). Pseudo-conservation laws in cyclic service systems. *J. Appl. Prob.* **24**, 949-964.

[43] Boxma, O.J., Groenendijk, W.P. (1988). Two queues with alternating service and switching times. In: *Queueing Theory and its Applications - Liber Amicorum for J.W. Cohen*, eds. O.J. Boxma, R. Syski (North-Holland, Amsterdam), 261-282.

[44] Boxma, O.J., Groenendijk, W.P. (1988). Waiting times in discrete-time cyclic-service systems. *IEEE Trans. Commun.* **36**, 164-170.

[45] Boxma, O.J., Groenendijk, W.P., Weststrate, J.A. (1990). A pseudo-conservation law for service systems with a polling table. *IEEE Trans. Commun.* **38**, 1865-1870.

[46] Boxma, O.J., Kelbert, M. (1994). Stochastic bounds for a polling system. *Ann. Oper. Res.* **48**, *Special Issue on Queueing Networks*, ed. N.M. van Dijk, 295-310.

[47] Boxma, O.J., Levy, H., Weststrate, J.A. (1990). Optimization of polling systems. In: *Proc. Performance '90*, eds. P.J.B. King, I. Mitrani, R.J. Pooley (North-Holland, Amsterdam), 349-361.

[48] Boxma, O.J., Levy, H., Weststrate, J.A. (1991). Efficient visit frequencies for polling tables: minimization of waiting cost. *Queueing Systems* **9**, 133-162.

[49] Boxma, O.J., Levy, H., Weststrate, J.A. (1993). Efficient visit orders for polling systems. *Perf. Eval.* **18**, 103-123.

[50] Boxma, O.J., Levy, H., Yechiali, U. (1992). Cyclic reservation schemes for efficient operation of multiple-queue single-server systems. *Ann. Oper. Res.* **35**, 187-208.

[51] Boxma, O.J., Meister, B.W. (1987). Waiting-time approximations for cyclic-service systems with switchover times. *Perf. Eval.* **7**, 299-308.

[52] Boxma, O.J., Meister, B.W. (1987). Waiting-time approximations in multi-queue systems with cyclic-service. *Perf. Eval.* **7**, 59-70.

[53] Boxma, O.J., Weststrate, J.A. (1989). Waiting times in polling systems with Markovian server routing. In: *Messung, Modellierung und Bewertung von Rechensystemen und Netzen*, eds. G. Stiege, J.S. Lie (Springer, Berlin), 89-104.

[54] Boxma, O.J., Weststrate, J.A., Yechiali, U. (1993). A globally gated polling system with server interruptions, and applications to the repairman problem. *Prob. Eng. Inf. Sc.* **7**, 187-208.

[55] Bozer, Y.A., Srinivasan, M.M. (1991). Tandem configurations for automated guided vehicle systems and the analysis of single-vehicle loops. *IIE Trans.* **23**, 72-82.

[56] Browne, S., Coffman, E.G. Jr., Gilbert, E.N., Wright, P.E.W. (1992). Gated, exhaustive, parallel service. *Prob. Eng. Inf. Sc.* **6**, 217-239.

[57] Browne, S., Kella, O. (1992). Parallel service with vacations. To appear in *Oper. Res.*

[58] Browne, S., Weiss, G. (1992). Dynamic priority rules when polling with multiple parallel servers. *Oper. Res. Lett.* **12**, 129-137.

[59] Browne, S., Yechiali, U. (1989). Dynamic priority rules for cyclic-type queues. *Adv. Appl. Prob.* **21**, 432-450.

[60] Brumelle, S.L. (1971). On the relation between customer and time averages in queues. *J. Appl. Prob.* **8**, 508-520.

[61] Bunday, B.D., El-Badri, W.K. (1988). The efficiency of M groups of N machines served by a travelling robot: comparison of two models. *Int. J. Prod. Res.* **26**, 299-308.

[62] Bux, W., Truong, H.L. (1983). Mean-delay approximations for cyclic-service queueing systems. *Perf. Eval.* **3**, 187-196.

[63] Buyukkoc, C., Varaiya, P., Walrand, J. (1985). The $c\mu$ rule revisited. *Adv. Appl. Prob.* **17**, 237-238.

[64] Buzacott, J.A., Shanthikumar, J.G. (1992). Design of manufacturing systems using queueing models. *Queueing Systems* **12**, *Special Issue on Queueing Models of Manufacturing Systems*, 135-213.

[65] Buzen, J.P., Chen, P.P.-S. (1974). Optimal load balancing in memory hierarchies. In: *Proc. IFIP 1974*, ed. J.L. Rosenfeld (North-Holland, Amsterdam), 271-275.

[66] Chang, K.C., Sandhu, D. (1992). Mean waiting time approximations in cyclic-service systems with exhaustive limited service policy. *Perf. Eval.* **15**, 21-40.

[67] Cheng, W.C., Muntz, R.R. (1990). Optimal routing for closed queueing networks. In: *Proc. Performance '90*, eds. P.J.B. King, I. Mitrani, R.J. Pooley (North-Holland, Amsterdam), 3-17.

[68] Choudhury, G.L. (1990). Polling with a general service order table: gated service. In: *Proc. INFOCOM '90*, 268-276.

[69] Coffman, E.G. Jr., Gilbert, E.N. (1986). A continuous polling system with constant service times. *IEEE Trans. Inform. Theory* **33**, 584-591.

[70] Coffman, E.G. Jr., Lueker, G.S., Rinnooy Kan, A.H.G. (1988). Asymptotic methods in the probabilistic analysis of sequencing and packing heuristics. *Mgmt. Sc.* **34**, 266-290.

[71] Coffman, E.G. Jr., Mitrani, I., Fayolle, G. (1988). Two queues with alternating service periods. In: *Proc. Performance '87*, eds. P.-J. Courtois, G. Latouche (North-Holland, Amsterdam), 227-239.

[72] Coffman, E.G. Jr., Stolyar, A. (1993). Continuous polling on graphs. *Prob. Eng. Inf. Sc.* **7**, 209-226.

[73] Cohen, J.W. (1982). *The Single Server Queue* (North-Holland, Amsterdam, 2nd ed.).

[74] Cohen, J.W., Boxma, O.J. (1981). The $M/G/1$ queue with alternating service formulated as a Riemann-Hilbert boundary value problem. In: *Proc. Performance '81*, ed. F.J. Kylstra (North-Holland, Amsterdam), 181-199.

[75] Cohen, J.W. (1988). A two-queue model with semi-exhaustive alternating service. eds. P.-J. Courtois, G. Latouche (North-Holland, Amsterdam), 19-37.

[76] Combé, M.B. (1994). Modelling dependence between interarrival and service times with Markovian arrival processes. CWI Report BS-R9412.

[77] Cooper, R.B., Murray, G. (1969). Queues served in cyclic order. *Bell Syst. Techn. J.* **48**, 675-689.

[78] Cooper, R.B. (1970). Queues served in cyclic order: waiting times. *Bell Syst. Techn. J.* **49**, 399-413.

[79] Cooper, R.B., Niu, S.-C., Srinivasan, M.M. (1992). A decomposition theorem for polling models: the switchover times are effectively additive. To appear in *Oper. Res.*

[80] De Souza e Silva, E., Gail, H.R., Muntz, R.R. (1994). Polling systems with server timeouts. Preprint.

[81] De Souza e Silva, E., Gerla, M. (1985). Load balancing in distributed systems with multiple classes and site constraints. In: *Proc. Performance '84*, ed. E. Gelenbe (North-Holland, Amsterdam), 17-33.

[82] Doshi, B.T. (1985). Analysis of clocked schedules - high priority tasks. *AT & T Techn. J.* **64**, 633-660.

[83] Eisenberg, M. (1971). Two queues with changeover times. *Oper. Res.* **19**, 386-401.

[84] Eisenberg, M. (1972). Queues with periodic service and changeover times. *Oper. Res.* **20**, 440-451.

[85] Eisenberg, M. (1979). Two queues with alternating service. *SIAM J. Appl. Math.* **36**, 287-303.

[86] Eisenberg, M. (1993). The polling system with a stopping server. Report AT & T Bell Laboratories, Holmdel, NJ.

[87] Everitt, D.E. (1986). A conservation-type law for the token ring with limited service. *Br. Telecom Techn. J.* **4**, 51-61.

[88] Everitt, D.E. (1986). Simple approximations for token rings. *IEEE Trans. Commun.* **34**, 719-721.

[89] Everitt, D.E. (1989). An approximation procedure for cyclic service queues with limited service. In: *Performance of Distributed and Parallel Systems*, eds. T. Hasegawa, H. Takagi, Y. Takahashi (North-Holland, Amsterdam), 141-156.

[90] Fabian, O., Levy, H. (1994). Pseudo-cyclic policies for multi-queue single server systems. *Ann. Oper. Res.* **48**, *Special Issue on Queueing Networks*, ed. N.M. van Dijk, 127-152.

[91] Ferguson, M.J. (1985). Mean waiting time for a token ring with nodal dependent overheads. In: *Teletraffic Issues in an Advanced Information Society, Proc. ITC-11*, ed. M. Akiyama (North-Holland, Amsterdam), 634-640.

[92] Ferguson, M.J. (1986). Mean waiting time for a token ring with station dependent overheads. In: *Local Area & Multiple Access Networks*, ed. R.L. Pickholtz (Computer Science Press, Rockville, MD), 43-67.

[93] Ferguson, M.J., Aminetzah, Y.J. (1985). Exact results for nonsymmetric token ring systems. *IEEE Trans. Commun.* **33**, 223-231.

[94] Fournier, L., Rosberg, Z. (1991). Expected waiting times in cyclic service systems under priority disciplines. *Queueing Systems* **9**, 419-439.

[95] Franken, P., König, D., Arndt, U., Schmidt, V. (1982). *Queues and Point Processes* (Wiley, New York).

[96] Fredericks, A.A., Farrell, B.L., DeMaio, D.F. (1985). Approximate analysis of a generalized clocked schedule. *AT & T Techn. J.* **64**, 597-615.

[97] Fricker, C., Jaïbi, M.R. (1994). Monotonicity and stability of periodic polling models. *Queueing Systems* **15**, 211-238.

[98] Fricker, C., Jaïbi, M.R. (1994). Stability of polling models with Markovian routing. Report INRIA Cedex 2278.

[99] Fuhrmann, S.W. (1981). Performance analysis of a class of cyclic schedules. Bell Laboratories technical memorandum 81-59531-1.

[100] Fuhrmann, S.W. (1985). Symmetric queues served in cyclic order. *Oper. Res. Lett.* **4**, 139-144.

[101] Fuhrmann, S.W. (1992). A decomposition result for a class of polling models. *Queueing Systems* **11**, *Special Issue on Polling Models*, 109-120.

[102] Fuhrmann, S.W., Cooper, R.B. (1985). Application of decomposition principle in $M/G/1$ vacation model to two continuum cyclic queueing models - especially token-ring LANs. *AT & T Techn. J.* **64**, 1091-1099.

[103] Fuhrmann, S.W., Cooper, R.B. (1985). Stochastic decompositions in the $M/G/1$ queue with generalized vacations. *Oper. Res.* **33**, 1117-1129.

[104] Fuhrmann, S.W., Wang, Y.T. (1988). Analysis of cyclic service systems with limited service: bounds and approximations. *Perf. Eval.* **9**, 35-54.

[105] Garey, M.R., Johnson, D.S. (1979). *Computers and Intractability: a Guide to the Theory of NP-Completeness* (Freeman, San Francisco).

[106] Gamse, B., Newell, G.F. (1982). An analysis of elevator operation in moderate height buildings - I. A single elevator. *Transp. Res. B* **16**, 303-319.

[107] Gamse, B., Newell, G.F. (1982). An analysis of elevator operation in moderate height buildings - II. Multiple elevators. *Transp. Res. B* **16**, 321-335.

[108] Georgiadis, L., Szpankowski, W. (1992). Stability of token passing rings. *Queueing Systems* **11**, *Special Issue on Polling Models*, 7-34.

[109] Gersht, A.M., Marbukh, V.V. (1975). Queueing systems with readjustment. *Engin. Cyb.* **13**, 55-65.

[110] Graham, R.L. (1966). Bounds for certain multiprocessing anomalies. *Bell Syst. Techn. J.* **45**, 1563-1581.

[111] Graham, R.L. (1969). Bounds on multiprocessing timing anomalies. *SIAM J. Appl. Math.* **17**, 263-269.

[112] Grillo, D. (1990). Polling mechanism models in communication systems - some application examples. In: *Stochastic Analysis of Computer and Communication Systems*, ed. H. Takagi (North-Holland, Amsterdam), 659-698.

[113] Groenendijk, W.P. (1989). Waiting-time approximations for cyclic service systems with mixed service strategies. In: *Teletraffic Science for New Cost-Effective Systems, Networks and Services, ITC 12*, ed. M. Bonatti (North-Holland, Amsterdam), 1434-1441.

[114] Groenendijk, W.P. (1990). *Conservation Laws in Polling Systems.* Ph.D. Thesis, University of Utrecht, Utrecht.

[115] Gupta, D., Srinivasan, M.M. (1993). When should a roving server be patient? Report University of Tennessee, Knoxville, TN.

[116] Hajek, B. (1985). Extremal splittings of point processes. *Math. Oper. Res.* **10**, 543-556.

[117] Harink, J.H.A., Cramer, P., Huitema, G.B. (1992). Optimization of polling call records from switches. Report TI-RA-92-435, PTT Research, Groningen.

[118] Hofri, M., Rosberg, Z. (1987). Packet delay under the Golden Ratio weighted TDM policy in a multiple-access channel. *IEEE Trans. Inform. Theory* **33**, 341-349.

[119] Hofri, M., Ross, K.W. (1987). On the optimal control of two queues with server set-up times and its analysis. *SIAM J. Comput.* **16**, 399-420.

[120] Itai, A., Rosberg, Z. (1984). A Golden Ratio control policy for a multiple-access channel. *IEEE Trans. Autom. Control* **29**, 712-718.

[121] Kamal, A.E., Hamacher, V.C. (1989). Approximate analysis of non-exhaustive multiserver polling systems with applications to local area networks. *Comput. Netw. ISDN Syst.* **17**, 15-27.

[122] Kao, E.P.C., Narayanan, K.S. (1991). Analyses of an $M/M/N$ queue with servers' vacations. *Eur. J. Oper. Res.* **54**, 256-266.

[123] Karmarkar, V.V., Kuhl, J.G. (1989). An integrated approach to distributed demand assignment in multiple-bus local networks. *IEEE Trans. Comput.* **38**, 679-695.

[124] Keilson, J., Servi, L.D. (1986). Oscillating random walk models for $GI/G/1$ vacation systems with Bernoulli schedules. *J. Appl. Prob.* **23**, 790-802.

[125] Keilson, J., Servi, L.D. (1990). The distributional form of Little's law and the Fuhrmann-Cooper decomposition. *Oper. Res. Lett.* **9**, 239-247.

[126] Khamisy, A., Altman, E., Sidi, M. (1992). Polling systems with synchronization constraints. *Ann. Oper. Res.* **35**, 231-267.

[127] Kim, W.B., Königsberg, E. (1987). The efficiency of two groups of N machines served by a single robot. *J. Oper. Res. Soc.* **38**, 523-538.

[128] Kleinrock, L., Levy, H. (1988). The analysis of random polling systems. *Oper. Res.* **36**, 716-732.

[129] Konheim, A.G., Levy, H. (1992). Efficient analysis of polling systems. In: *Proc. INFOCOM '92*, 2325-2331.

[130] Königsberg. E., Mamer, J. (1982). The analysis of production systems. *Int. J. Prod. Res.* **20**, 1-16.

[131] Koole, G.M. (1994). Assigning a single server to inhomogeneous queues with switching costs. CWI Report BS-R9405.

[132] Kroese, D.P., Schmidt, V. (1992). A continuous polling system with general service times. *Ann. Appl. Prob.* **2**, 906-927.

[133] Kroese, D.P., Schmidt, V. (1992). Single-server queues with spatially distributed arrivals. Preprint.

[134] Kruskal, J.B. (1969). Work-scheduling algorithms: a non-probabilistic queueing study (with possible applications to No. 1 ESS). *Bell Syst. Techn. J.* **48**, 2963-2974.

[135] Kühn, P.J. (1979). Multiqueue systems with nonexhaustive cyclic service. *Bell Syst. Techn. J.* **58**, 671-698.

[136] Lee, D.-S., Sengupta, B. (1992). A reservation based cyclic server queue with limited service. In: *Perf. Eval. Review* **20**, 70-77 (Proc. 1992 ACM SIGMETRICS & Performance '92).

[137] Lee, D.-S., Sengupta, B. (1992). An approximate analysis of a cyclic server queue with limited service and reservations. *Queueing Systems* **11**, 153-178.

[138] Levy, H. (1988). Optimization of polling systems: the fractional exhaustive service method. Report Tel Aviv University, Tel Aviv.

[139] Levy, H. (1989). Analysis of cyclic polling systems with binomial gated service. In: *Performance of Distributed and Parallel Systems*, eds. T. Hasegawa, H. Takagi, Y. Takahashi (North-Holland, Amsterdam), 127-139.

[140] Levy, H., Kleinrock, L. (1991). Polling systems with zero switch-over periods: a general method for analyzing the expected delay. *Perf. Eval.* **13**, 97-107.

[141] Levy, H., Sidi, M. (1990). Polling systems: applications, modelling and optimization. *IEEE Trans. Commun.* **38**, 1750-1760.

[142] Levy, H., Sidi, M., Boxma, O.J. (1990). Dominance relations in polling systems. *Queueing Systems* **6**, 155-171.

[143] Levy, Y., Yechiali, U. (1976). An $M/M/s$ queue with servers' vacations. *INFOR* **14**, 153-163.

[144] Loucks, W.M., Hamacher, V.C., Preiss, B.R., Wong, L. (1985). Short-packet transfer performance in local area ring networks. *IEEE Trans. Comput.* **34**, 1006-1014.

[145] Liu, Z., Nain, P., Towsley, D. (1992). On optimal polling policies. *Queueing Systems* **11**, *Special Issue on Polling Models*, 59-83.

[146] Lucantoni, D.M. (1991). New results on the single server queue with a batch Markovian arrival process. *Commun. Stat. - Stoch. Mod.* **7**, 1-46.

[147] Lucantoni, D.M. (1993). The $BMAP/G/1$ queue: a tutorial. In: *Models and Techniques for the Performance Evaluation of Computer and Communication Systems*, eds. L. Donatiello, R. Nelson (Springer, Berlin), 330-358.

[148] Mack, C., Murphy, T., Webb, N.L. (1957). The efficiency of N machines unidirectionally patrolled by one operative when walking times and repair times are constants. *J. Roy. Stat. Soc. B* **19**, 166-172.

[149] Medhi, J. (1991). *Stochastic Models in Queueing Theory* (Academic Press, San Diego).

[150] Mei, R.D., van der, Borst, S.C. (1994). Analysis of multiple-server polling systems by means of the power-series algorithm. CWI Report BS-R9410. Submitted to *Perf. Eval.*

[151] Meilijson, I., Yechiali, U. (1977). On optimal right-of-way policies at a single-server station when insertion of idle times is permitted. *Stoch. Proc. Appl.* **6**, 25-32.

[152] Mitrany, I.L., Avi-Itzhak, B. (1968). A many-server queue with server interruptions. *Oper. Res.* **16**, 628-638.

[153] Morris, R.J.T., Wang, Y.T. (1984). Some results for multi-queue systems with multiple cyclic servers. In: *Performance of Computer-Communication Systems*, eds. W. Bux, H. Rudin (North-Holland, Amsterdam), 245-258.

[154] Neuts, M.F. (1989). *Structured Stochastic Matrices of $M/G/1$ Type and Their Applications* (Marcel Dekker, New York).

[155] Neuts, M.F., Lucantoni, D.M. (1979). A Markovian queue with N servers subject to breakdowns and repairs. *Mgmt. Sc.* **25**, 849-861.

[156] Panwar, S.S., Philips, T.K., Chen, M.-S. (1992). Golden Ratio scheduling for flow control with low buffer requirements. *IEEE Trans. Commun.* **40**, 765-772.

[157] Raith, T. (1985). Performance analysis of multibus interconnection networks in distributed systems. In: *Teletraffic Issues in an Advanced Information Society, Proc. ITC-11*, ed. M. Akiyama (North-Holland, Amsterdam), 662-668.

[158] Resing, J.A.C. (1990). *Asymptotic Results in Feedback Systems.* Ph.D. Thesis, Technical University Delft, Delft.

[159] Resing, J.A.C. (1993). Polling systems and multitype branching processes. *Queueing Systems* **13**, 409-426.

[160] Sarkar, D., Zangwill, W.I. (1989). Expected waiting time for nonsymmetric cyclic queueing systems - Exact results and applications. *Mgmt. Sc.* **35**, 1463-1474.

[161] Sarkar, D., Zangwill, W.I. (1991). Variance effects in cyclic production systems. *Mgmt. Sc.* **37**, 444-453.

[162] Schassberger, R. (1994). Stability of polling networks with state-dependent server routing. Preprint.

[163] Seneta, E. (1981). *Non-negative Matrices and Markov Chains* (Springer, New York, 2nd ed.).

[164] Servi, L.D. (1986). Average delay approximation of $M/G/1$ cyclic service queue with Bernoulli schedule. *IEEE J. Sel. Areas Commun.* **4**, 813-822.

[165] Shimogawa, S., Takahashi, Y. (1992). A note on the pseudo-conservation law for a multi-queue with local priority. *Queueing Systems* **11**, *Special Issue on Polling Models*, 145-151.

[166] Sidi, M., Levy, H. (1990). Customer routing in polling systems. In: *Proc. Performance '90*, eds. P.J.B. King, I. Mitrani, R.J. Pooley (North-Holland, Amsterdam), 319-331.

[167] Sidi, M., Levy, H., Fuhrmann, S.W. (1992). A queueing network with a single cyclically roving server. *Queueing Systems* **11**, *Special Issue on Polling Models*, 121-144.

[168] Srinivasan, M.M. (1988). An approximation for mean waiting times in cyclic server systems with nonexhaustive service. *Perf. Eval.* **9**, 17-33.

[169] Srinivasan, M.M., Niu, S.-C., Cooper, R.B. (1993). Relating polling models with nonzero and zero switchover times. Report University of Tennessee, Knoxville, TN.

[170] Stadje, W. (1985). The busy period of the queueing system $M/G/\infty$. *J. Appl. Prob.* **22**, 697-704.

[171] Takács, L. (1968). Two queues attended by a single server. *Oper. Res.* **16**, 639-650.

[172] Takagi, H. (1986). *Analysis of Polling Systems* (The MIT Press, Cambridge, MA).

[173] Takagi, H. (1991). Application of polling models to computer networks. *Comp. Netw. ISDN Syst.* **22**, 193-211.

[174] Takagi, H. (1991). *Queueing Analysis, Vol. 1* (North-Holland, Amsterdam).

[175] Takagi, H. (1990). Queueing analysis of polling models: an update. In: *Stochastic Analysis of Computer and Communication Systems*, ed. H. Takagi (North-Holland, Amsterdam), 267-318.

[176] Takagi, H. (1994). Queueing analysis of polling models: progress in 1990-1993. To appear in: *Frontiers in Queueing: Models, Methods and Problems*, ed. J.H. Dshalalow (CRC Press).

[177] Takine, T., Hasegawa, T. (1992). On the $M/G/1$ queue with multiple vacations and gated service discipline. *JORSJ* **35**, 217-235.

[178] Tantawi, A.N., Towsley, D. (1985). A general model for optimal static load balancing in star network configurations. In: *Proc. Performance '84*, ed. E. Gelenbe (North-Holland, Amsterdam), 277-291.

[179] Tantawi, A.N., Towsley, D. (1985). Optimal static load balancing in distributed computer systems. *J. ACM* **32**, 445-465.

[180] Tripathi, S.K., Woodside, C.M. (1988). A vertex-allocation theorem for resources in queueing networks. *J. ACM* **35**, 221-230.

[181] Tedijanto, T.E. (1990). *Nonexhaustive policies in polling systems and vacation models.* Ph.D. Thesis, University of Maryland, College Park, MD.

[182] Tedijanto, T.E. (1991). Stochastic comparisons in vacation models. *Commun. Stat. - Stoch. Mod.* **7**, 125-135.

[183] Tedijanto, T.E. (1992). A note on the comparison between Bernoulli and limited policies in vacation models. *Perf. Eval.* **15**, 89-97.

[184] Tijms, H.C. (1986). *Stochastic Modelling and Analysis: a Computational Approach* (Wiley, Chichester).

[185] Titchmarsh, E.C. (1939). *The Theory of Functions* (Oxford University Press, London, 2nd ed.).

[186] Wang, Y.-T., Morris, R.J.T. (1985). Load sharing in distributed systems. *IEEE Trans. Comput.* **34**, 204-217.

[187] Watson, K. S. (1985). Performance evaluation of cyclic service strategies - a survey. In: *Proc. Performance '84*, ed. E. Gelenbe (North-Holland, Amsterdam), 521-533.

[188] Weststrate, J.A. (1992). *Analysis and Optimization of Polling Models.* Ph.D. Thesis, University of Tilburg, Tilburg.

[189] Woodside, C.M., Tripathi, S.K. (1986). Optimal allocation of file servers in a local network environment. *IEEE Trans. Softw. Eng.* **12**, 844-848.

[190] Yang, Q., Ghosal, D., Bhuyan, L.N. (1986). Performance analysis of multiple token ring and multiple slotted ring networks. In: *Proc. 1986 Comput. Netw. Symp.*, 79-86.

[191] Yechiali, U. (1991). Optimal dynamic control of polling systems. In: *Queueing, Performance and Control in ATM*, eds. J.W. Cohen, C.D. Pack (North-Holland, Amsterdam), 205-217.

[192] Zafirovic-Vukotic, M., Niemegeers, I.G., Valk, D.S. (1988). Performance modelling of slotted ring protocols in HSLAN's. *IEEE J. Sel. Areas Commun.* **6**, 1001-1024.

Samenvatting (Summary)

Dit proefschrift is gewijd aan de analyse en optimalisering van zogenaamde polling systemen. Het klassieke polling model bestaat uit een aantal wachtrijen waar klanten arriveren die bediening verlangen door één enkele gemeenschappelijke bediende die de wachtrijen volgens een of ander cyclisch schema bezoekt. De zogeheten *bedieningsdiscipline* specificeert welke klanten de bediende tijdens een bezoek aan een wachtrij dient te helpen, terwijl de zogeheten *routeringsdiscipline* bepaalt in welke volgorde de bediende de wachtrijen dient te bezoeken. Doorgaans wordt verondersteld dat het omschakelen van de ene wachtrij naar de andere een zekere omschakeltijd vergt. Polling modellen dienen om congestieverschijnselen te bestuderen zoals die zich voordoen in situaties waarin gebruikers, behorend tot een aantal verschillende klassen, gelijktijdig bediening verlangen door één enkele gemeenschappelijke bedieningsfaciliteit. In de loop der jaren hebben polling modellen talrijke toepassingen gevonden in de prestatie-analyse van computersystemen, communicatienetwerken, productiesystemen, en onderhoudsstrategieën.
Nu volgt een beknopt overzicht van de inhoud van de diverse hoofdstukken van het proefschrift.
In hoofdstuk 1 beschrijven we het klassieke polling model tezamen met de voornaamste varianten, en bespreken we de belangrijkste resultaten op het gebied van de analyse en optimalisering van polling systemen.
In hoofdstuk 2 gaan we in op het gebruik van decompositie-eigenschappen bij het analyseren van polling modellen, speciaal aandacht schenkend aan het bestaan van zogenaamde pseudo-behoudswetten voor de gemiddelde wachttijden. Deze decompositie-eigenschappen relateren prestatiematen zoals de hoeveelheid werk en de rijlengten in een systeem mèt omschakeltijden aan dezelfde prestatiemaat in een overeenkomstig systeem zònder omschakeltijden. Verder demonstreren we het gebruik van dergelijke decompositie-eigenschappen bij het bestuderen van verwante modellen waarin niet de bediende de wachtrijen bezoekt om daar de eventueel aanwezige klanten te helpen maar een verzamelaar van tijd tot tijd de wachtrijen bezoekt, de eventueel aanwezige klanten ophaalt, en vervolgens aflevert bij de bediende om daar te worden geholpen.
In hoofdstuk 3 beschouwen we twee verschillende maar toch verwante polling modellen: (i) een model zònder omschakeltijden, en (ii) een model mèt omscha-

keltijden, waarin de bediende blijft omschakelen wanneer het systeem leeg is geraakt. Voor beide modellen relateren we de marginale rijlengteverdeling bij een wachtrij op een willekeurig moment aan de rijlengteverdeling op momenten waarop bezoeken beginnen en eindigen bij die wachtrij. Als nevenresultaat verkrijgen we een aanmerkelijk korter bewijs van de zogenaamde Fuhrmann-Cooper rijlengtedecompositie. Voor de klasse van bedieningsdisciplines met een vertakkingsstructuur leggen we een nauwe relatie tussen zowel de rijlengte- als de wachttijdverdeling in beide modellen. Verder laten we zien hoe deze relatie kan worden benut om de complexiteit van de berekening van de gemidelde wachttijden drastisch te reduceren.

In de hoofdstukken 4 en 5 bestuderen we polling systemen waarin het de bediende is toegestaan halt te houden bij een wachtrij wanneer zich geen klanten in het systeem bevinden. Allereerst leiden we in hoofdstuk 4 een pseudo-behoudswet af voor een algemeen model, ruimte latend voor een verscheidenheid aan bedieningsdisciplines, waarin het de bediende is toegestaan halt te houden bij een willekeurige deelverzameling van de wachtrijen. We gebruiken de pseudo-behoudswet vervolgens om het geval van een halt-houdende bediende te vergelijken met dat van een *niet*-halt-houdende bediende. Verder onderzoeken we bij welke wachtrijen de bediende halt dient te houden om de gemiddelde totale hoeveelheid werk in het systeem te minimaliseren.

In hoofdstuk 5 richten we de aandacht op een model met de zogenaamde *globally gated* bedieningsdiscipline, waarin de bediende halt houdt op zijn thuisbasis wanneer zich geen klanten in het systeem bevinden. We leiden expliciete uitdrukkingen af voor de Laplace-Stieltjes getransformeerde van de cyclustijdverdeling en van de wachttijdverdeling bij elk van de wachtrijen en tevens voor de kansgenererende functie van de gezamenlijke rijlengteverdeling op polling momenten. Verder tonen we aan dat in het geval van een halt-houdende bediende de wachttijd bij elk van de wachtrijen kleiner is dan in dat van een *niet*-halt-houdende bediende.

De hoofdstukken 6 en 7 zijn gewijd aan verschillende optimaliseringsproblemen in polling systemen. Allereerst beschouwen we in hoofdstuk 6 polling systemen met de zogenaamde *k-limited* bedieningsdiscipline. Onder de k-limited bedieningsdiscipline werkt de bediende tijdens een bezoek aan een wachtrij Q_i totdat òf een vooraf gespecificeerd aantal van k_i klanten is bediend òf de wachtrij leeg is geraakt, afhankelijk van wat zich het eerst voordoet. We zijn geïnteresseerd in het bepalen van geschikte waarden voor de k_i's die bijdragen tot een efficiënte werking van het systeem. We ontwikkelen een heuristische aanpak van het probleem, welke uitvoerig door middel van numerieke experimenten wordt getest.

In hoofdstuk 7 beschouwen we polling systemen met een zogenaamd *fixed time polling* (ftp) schema. Een ftp schema specificeert welke wachtrij de bediende wanneer dient te bezoeken, d.w.z., het specificeert niet alleen de *bezoekvolgorde*, maar tevens de *bezoektijden*. We zijn geïnteresseerd in het construeren van ftp schema's die bijdragen tot een efficiënte werking van het systeem. Uitgaande van tamelijk eenvoudige wachttijdbenaderingen, formuleren we het

probleem als een mathematisch programmeringsprobleem. Wegens de NP-lastigheid ontwikkelen we vervolgens een heuristische methode om het mathematisch programmeringsprobleem op te lossen, welke uitgebreid door middel van numerieke experimenten wordt getest. Het betreffende onderzoek is uitgevoerd in nauwe samenwerking met PTT Research, Groningen.
De voorgaande 7 hoofdstukken hebben betrekking op de analyse en optimalisering van polling systemen met één enkele bediende. In de overigens uitgebreide polling literatuur is tot dusverre nauwelijks aandacht geschonken aan polling systemen met meerdere bedienden, hoewel dergelijke modellen op natuurlijke wijze optreden in de prestatie-analyse van bijvoorbeeld gespreide computersystemen, slotted-ring communicatienetwerken, rotondes, en liften. De volgende 3 hoofdstukken zijn gewijd aan dergelijke modellen met meerdere wachtrijen en meerdere bedienden. Allereerst beschouwen we in hoofdstuk 8 een systeem bestaande uit meerdere klantentypen die door meerdere niet-identieke parallelle bedienden dienen te worden geholpen. We zijn geïnteresseerd in het vinden van een optimale probabilistische toewijzing van de klantentypen aan de bedienden. We karakteriseren de structuur van een optimale toewijzing en beschrijven vervolgens voor enkele speciale gevallen in detail hoe de structuur kan worden benut bij het bepalen van een optimale toewijzing.
In de hoofdstukken 9 en 10 bestuderen we polling modellen met meerdere bedienden, waarin de samenwerking, anders dan in hoofdstuk 8, resulteert in feitelijke interactie. Tot dusverre zijn nauwelijks exacte resultaten bekend voor dergelijke polling modellen, afgezien van enkele gemiddelde-waarde resultaten voor globale prestatiematen zoals cyclustijden. Allereerst onderzoeken we in hoofdstuk 9 systemen waarin de bedienden worden verondersteld te zijn *gekoppeld*, d.w.z., de bedienden bezoeken de wachtrijen gelijktijdig. Voor de klasse van systemen die een exacte analyse toelaten leiden we uitdrukkingen af voor de wachttijdverdeling, de marginale rijlengteverdeling op willekeurige momenten, en voor de gezamenlijke rijlengteverdeling op polling momenten. Deze klasse van systemen omvat (i) verscheidene systemen met één enkele wachtrij en een variabel aantal bedienden; (ii) systemen met twee wachtrijen, twee bedienden, de exhaustive bedieningsdiscipline, en exponentiële bedieningsduren; en (iii) systemen met een oneindig aantal bedienden, een willekeurig aantal wachtrijen, de exhaustive of gated bedieningsdiscipline, en deterministische bedieningsduren.
In hoofdstuk 10 richten we de aandacht op systemen waarin de bedienden worden verondersteld *onafhankelijk* te zijn, d.w.z., elk van de bedienden bezoekt de wachtrijen volgens zijn eigen cyclische schema. Aangezien deze klasse van systemen een exacte analyse volledig uitsluit, ontwikkelen we wachttijdbenaderingen voor het geval van de exhaustive en gated bedieningsdiscipline. Met behulp van numerieke experimenten worden de benaderingen getest voor een verscheidenheid aan parametercombinaties.